

Iterative Channel Estimation Using LSE and Sparse Message Passing for MmWave MIMO Systems

Chongwen Huang, *Student Member, IEEE*, Lei Liu, *Student Member, IEEE*,
Chau Yuen, *Senior Member, IEEE*, Sumei Sun, *Fellow, IEEE*,

Abstract

We propose an iterative channel estimation algorithm based on the Least Square Estimation (LSE) and Sparse Message Passing (SMP) algorithm for the Millimeter Wave (mmWave) MIMO systems. The channel coefficients of the mmWave MIMO are approximately modeled as a Bernoulli-Gaussian distribution since there are relatively fewer paths in the mmWave channel, i.e., the channel matrix is sparse and only has a few non-zero entries. By leveraging the advantage of sparseness, we proposed an algorithm that iteratively detects the exact location and value of non-zero entries of the sparse channel matrix. The SMP is used to detect the exact location of non-zero entries of the channel matrix, while the LSE is used for estimating its value at each iteration. We also analyze the Cramer-Rao Lower Bound (CRLB), and show that the proposed algorithm is a minimum variance unbiased estimator. Furthermore, we employ the Gaussian approximation for message densities under density evolution to simplify the analysis of the algorithm, which provides a simple method to predict the performance of the proposed algorithm. Numerical experiments show that the proposed algorithm has much better performance than the existing sparse estimators, especially when the channel is sparse. In addition, our proposed algorithm converges to the CRLB of the genie-aided estimation of sparse channels in just 5 turbo iterations.

Chau Yuen and Chongwen Huang are with the Singapore University of Technology and Design, Singapore (e-mail: yuenchau@sutd.edu.sg, Chongwen_huang@mymail.sutd.edu.sg).

Lei Liu is with the State Key Lab of Integrated Services Networks, Xidian University, Xian, 710071, China (e-mail: lliu_0@stu.xidian.edu.cn).

Sumei Sun is with the Institute for Infocomm Research (I2R), Agency for Science, Technology and Research (A*STAR), 138632, Singapore (e-mail: sunsm@i2r.a-star.edu.sg).

The material in this paper will be presented in part at the conference of IEEE Global Communications Washington, DC USA, Dec. 2016 [1].

Index Terms

Millimeter wave, iterative channel estimation, sparse message passing, Gaussian-Bernoulli distribution, Cramer-Rao lower bound, minimum variance unbiased estimator, Gaussian approximation.

I. INTRODUCTION

Millimeter wave (mmWave) is receiving tremendous interest by the academia, industry, and government for future 5G cellular systems [2], [3], [5], [4], [6]. The main reason is that the majority of our current wireless communications systems operating in the microwave spectrum (i.e., < 6 GHz) are by now crowded with the limited bandwidth. MmWave can take full advantage of spectrum from 30 GHz to 300 GHz and provide beyond 2 GHz of bandwidth [3], [4]. However, the large communication spectrum poses new challenges. One of the main challenges is that mmWave signal propagation is impaired by severe path-loss. Recent urban model experiments show that path losses are 40 dB worse at 28 GHz compared to 2.8 GHz [7], [8].

One way to overcome this severe path-loss of mmWave signal propagation is to improve beamforming gain by the increasing number of transmit antennas and receive antennas [6], [9], [10]. This means that the transmitter and receiver can focus more energy on the dominant propagation paths. Therefore, MmWave channels have fewer paths. This results in that mmWave channels are very sparse in both the angle and time domains [6], [11], [12], [10], [13]. Recent research measurements show that mmWave channels typically exhibit only 3-4 scattering clusters in dense-urban non-line-of-sight environments [7], [8], [14]. Therefore, conventional MIMO channel estimation methods cannot be applied directly in mmWave systems. For example, the message-passing based iterative channel estimation algorithms in [15], [16], [17], [18], [19], [20], which were designed for massive MIMO systems. Even though these algorithms have reduced the complexity, they were not suitable for mmWave channels, as they did not account for the sparsity of channels. This prompts the need to design efficient channel estimation techniques for the mmWave systems.

For the sparse channel estimation, there are several algorithms that have been proposed in [26], [21], [24], [25], [22], [27], [23]. They can be classified into three categories according to how much priori information they need. The algorithm in the first category requires to know the full knowledge of the channel's distribution, structure, and the noise variance. For example, the approximate message passing (AMP) algorithm [21] proposed by Donoho and Maleki. It is a low-complexity iterative Bayesian algorithm that can achieve approximately maximum a posteriori and minimum mean-squared error signal estimates. Iterative Detection/Estimation With Threshold (ITD-SE) [22] and Adaptive Compressed Sensing (ACS) estimation algorithm proposed in [23] are classified into the second category, since they only need to know the partial priori information of the channel, e.g., the degree of sparsity L . ITD-SE based Least Square Estimation (LSE) needs

TABLE I
COMPARISON OF RELATED WORK OF SPARSE CHANNEL ESTIMATION

| Techniques | Need the priori knowledge? | Complexity | Iterations (typical) | Characteristics |
|---------------|----------------------------|--------------------|----------------------|--|
| AMP[21] | Full | $\mathcal{O}(N^2)$ | ≥ 25 | Iterative Bayesian algorithm, achieves approximately the minimum mean-squared error. |
| OMP[24] | No | $\mathcal{O}(N^3)$ | ≥ 50 | Complexity intensive, depends on the dictionary matrix. |
| SBL[25] | No | $\mathcal{O}(N^3)$ | ≥ 50 | Noise sensitive, complexity intensive. |
| LASSO[26] | No | $\mathcal{O}(N^2)$ | ≥ 25 | Needs to tune a parameter, trade-off between complexity and performance. |
| EM-BG-AMP[27] | No | $\mathcal{O}(N^2)$ | ≥ 25 | Noise sensitive, low complexity. |
| IDT-SE[22] | Partial | $\mathcal{O}(N^3)$ | ≥ 3 | Performance depends on the adaptive threshold selection. |
| ACS[23] | Partial | $\mathcal{O}(N^3)$ | L | Complexity intensive, depends on the designed high multi-resolution codebook. |
| Proposed | Partial | $\mathcal{O}(N^3)$ | 3-6 | Reaches CRLB and achieves the best performance when the channel is sparse. |

the fewer iterations, but its performance depends on the adaptive threshold selection scheme. ACS leverages the advanced compressed sensing theory and combines with the hybrid beamforming technique. Therefore, it is very suitable for the mmWave. The algorithms in the third category do not need know any priori knowledge of the channel, but these algorithms usually spend the more time to learn the channel for improving the estimation performance, for example, Orthogonal Matching Pursuit (OMP) [24] and Sparse Bayesian Learning [25]. Although LASSO [26] and EM-GM-AMP [27] are low complexity, the solution of LASSO is generally not the sparsest one, and EM-GM-AMP is the sensitive to noise especially at the high Signal Noise Ratio (SNR)[28]. The comparison among our work and previous work are listed in Table I, where CRLB denotes Cramer-Rao Lower Bound and N denotes the higher dimension of the row and column vector of the channel matrix.

In this paper, we develop an iterative channel estimation algorithm based on the LSE and Sparse Message Passing (SMP) algorithm for mmWave MIMO systems with large antenna arrays at both the transmitters and receivers. Then, a virtual channel representation model is adopted since it can capture the sparseness of physical mmWave modeling and provides simple geometric interpretation of the scattering environment ([see [12], [11]]). Based on this model, we further approximate the mmWave channel as a Bernoulli-Gaussian distribution. The main contributions of this paper are summarized as follows.

- We formulated a sparse channel estimation problem and proposed a novel sparse channel estimation algorithm. Compared with previously proposed sparse channel estimation, ours

can yield a better performance since it not only can take full advantage of the inherent sparseness of mmWave channel, but also can leverage both virtues of the LSE and SMP algorithms.

- We gave the performance analysis of the proposed algorithm, and derived its upper bound and CRLB. Furthermore, we showed that the proposed algorithm is Minimum Variance Unbiased Estimator (MVUE) under the assumption that there is the partial priori knowledge of the channel.
- We proposed a powerful analysis technique, which can provide significant insights on the convergence behavior on the iteration evolution of the proposed algorithm. This analysis used a Gaussian approximation for message densities under density evolution, and adopted the Log-Likelihood Ratios (LLRs) of messages that can simplify the analysis.
- We evaluated the performance of the proposed estimation algorithm. Numerical simulations show that our algorithm exhibits far better performance than the classical LSE estimator, as well as existing sparse estimators (e.g., LASSO, IED-SE, EM-BG-AMP, etc.). In addition, we also find that this algorithm needs only 5 turbo iterations to achieve the CRLB.

The rest of the paper is organized as follows. In Section II, we present the virtual channel representation model for mmWave MIMO systems and formulate a sparse channel estimation problem. In Section III, we propose an iterative sparse channel estimation algorithm. The performance analysis of the proposed algorithm is given in Section IV. In Section V, simulation results demonstrating the performance of the proposed algorithms are given, before concluding the paper in Section VI.

Notation: a is a scalar, \mathbf{a} is a vector and \mathbf{A} is a matrix. \mathbf{A}^T , \mathbf{A}^H , \mathbf{A}^{-1} , \mathbf{A}^\dagger and $\|\mathbf{A}\|_F$ represent transpose, Hermitian (conjugate transpose), inverse, pseudo-inverse and Frobenius norm of a matrix \mathbf{A} , respectively. $\mathbf{A} \otimes \mathbf{B}$ denotes the Kronecker product of \mathbf{A} and \mathbf{B} , and $\text{vec}(\mathbf{A})$ is a vector stacking all the columns of \mathbf{A} . $\text{diag}(\mathbf{a})$ is a diagonal matrix with the entries of \mathbf{a} on its diagonal, and $\text{diag}(\mathbf{A})$ is a block diagonal matrix with the matrix \mathbf{A} as the block on its diagonal. $\mathcal{N}(\mathbf{m}, \mathbf{V})$ is a complex Gaussian random vector with mean \mathbf{m} and covariance \mathbf{V} . We use the $E\{\cdot\}$ and $\text{Var}\{\cdot\}$ to denote the expectation and variance operation respectively. In addition, $E\{a|b\}$ denotes the conditional expectation of variable a given b , and $\text{Var}\{a|b\}$ denotes the conditional variance of the variable a given b .

II. SYSTEM MODEL

We consider a mmWave communication system that has N_t transmit antennas, N_r receive antennas, and the narrow band baseband received signal can be written as follows

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \quad (1)$$

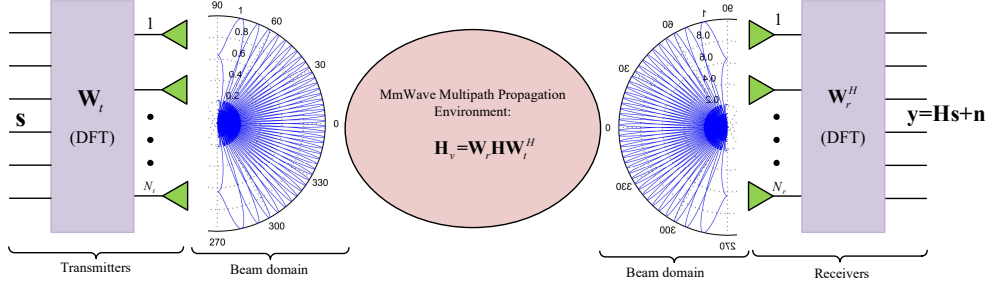


Fig. 1. Virtual channel representation of the mmWave multipath propagation environment. The Fourier transformation \mathbf{W}_t and \mathbf{W}_r can be seen as a mapping from the antenna domain onto a beam domain and the entries of the matrix \mathbf{H}_v can be interpreted as the channel gains between the N_t transmit and the N_r receive beams.

where $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$ is the channel matrix, $\mathbf{s} \in \mathbb{C}^{N_t \times 1}$ is the transmitted signal, $\mathbf{y} \in \mathbb{C}^{N_r \times 1}$ is the received signal and $\mathbf{n} \in \mathbb{C}^{N_r \times 1}$ is the Gaussian noise corrupting the received signal.

Since mmWave channels are expected to have limited scattering, we adopt a geometric channel model with L scatterers. Each scatterer is further assumed to contribute a single propagation path between transmitters and receivers [7], [23], [28]. Under this model, the channel \mathbf{H} can be expressed as

$$\mathbf{H} = \sum_{l=1}^L \alpha_l \mathbf{a}_r(\theta_l) \mathbf{a}_t^H(\phi_l), \quad (2)$$

where α_l is the gain of the l th path, $\phi_l \in [0, 2\pi]$ and $\theta_l \in [0, 2\pi]$ denote the l th path's azimuth angles of departure and arrival of transmitters and receivers respectively. Finally, $\mathbf{a}_t(\phi_l)$ and $\mathbf{a}_r(\theta_l)$ are the antenna array response vectors at transmitters and receivers respectively [23], [29], [30], [31]. If A Uniform Planar Array (UPA) is used, $\mathbf{a}_t(\phi_l)$ can be written as

$$\mathbf{a}_t(\phi_l) = \frac{1}{\sqrt{N_t}} \left[1, e^{j \frac{2\pi}{\lambda} \sin(\phi_l)}, \dots, e^{j (N_t - 1) \frac{2\pi}{\lambda} \sin(\phi_l)} \right]^T, \quad (3)$$

where λ is the signal wavelength, and d is the distance between antenna elements. The array response vectors at the receivers, $\mathbf{a}_r(\theta_l)$, can be written in a similar fashion. Then, the channel can be written in a more compact form as

$$\mathbf{H} = \mathbf{A}_r \text{diag}(\boldsymbol{\alpha}) \mathbf{A}_t^H, \quad (4)$$

where $\boldsymbol{\alpha} = \sqrt{\frac{N_r N_t}{\rho}} [\alpha_1, \alpha_2, \dots, \alpha_L]^T$. The matrices

$$\mathbf{A}_t = [\mathbf{a}_t(\phi_1), \mathbf{a}_t(\phi_2), \dots, \mathbf{a}_t(\phi_L)], \quad (5)$$

and

$$\mathbf{A}_r = [\mathbf{a}_r(\theta_1), \mathbf{a}_r(\theta_2), \dots, \mathbf{a}_r(\theta_L)], \quad (6)$$

contain the transmitters and receivers array response vectors. The mmWave multipath propagation channel usually consists of the line-of-sight (LOS) and a few reflected path clusters [14], [28], [31], [32], [33]. Large antenna arrays are deployed to get beamforming gain in order to combat the high path loss. Hence, we usually have $L \ll \min\{N_r, N_t\}$. We adopt a virtual channel representation that keeps the sparseness of mmWave modeling without its complexity, provides a tractable linear channel characterization, and offers a simple and transparent interpretation of the effects of scattering and array characteristics on channel capacity and diversity [11], [34], [35], [36]. The finite dimensionality of the signal space allows the virtual channel model that can be expressed as

$$\begin{aligned} \mathbf{H}_v &= \mathbf{W}_r^H \mathbf{H} \mathbf{W}_t \\ &= \underbrace{\left[\mathbf{a}_r\left(\frac{-\tilde{N}_r}{N_r}\right), \mathbf{a}_r\left(\frac{-\tilde{N}_r+1}{N_r}\right), \dots, \mathbf{a}_r\left(\frac{\tilde{N}_r}{N_r}\right) \right]^H}_{\mathbf{W}_r^H} \mathbf{H} \underbrace{\left[\mathbf{a}_t\left(\frac{-\tilde{N}_t}{N_t}\right), \mathbf{a}_t\left(\frac{-\tilde{N}_t+1}{N_t}\right), \dots, \mathbf{a}_t\left(\frac{\tilde{N}_t}{N_t}\right) \right]}_{\mathbf{W}_t} \end{aligned} \quad (7)$$

where $\tilde{N}_r := \frac{N_r-1}{2}$ and $\tilde{N}_t := \frac{N_t-1}{2}$ (assuming the N_r and N_t are odd.), and $\mathbf{H}_v \in \mathbb{C}^{N_r \times N_t}$ is no longer diagonal. Note that $\mathbf{W}_r \in \mathbb{C}^{N_r \times N_r}$ and $\mathbf{W}_t \in \mathbb{C}^{N_t \times N_t}$ are channel-invariant unitary DFT matrices [34], [13], and $\mathbf{W}_t \mathbf{W}_t^H = \mathbf{I}_{N_t}$, $\mathbf{W}_r \mathbf{W}_r^H = \mathbf{I}_{N_r}$. We recast (1) by the virtual channel representation as

$$\begin{aligned} \mathbf{y} &= \mathbf{W}_r \underbrace{\mathbf{W}_r^H \mathbf{H} \mathbf{W}_t}_{:=\mathbf{H}_v} \mathbf{W}_t^H \mathbf{s} + \mathbf{n} \\ &= \mathbf{W}_r \mathbf{H}_v \mathbf{W}_t^H \mathbf{s} + \mathbf{n}, \end{aligned} \quad (8)$$

The Fourier transformation \mathbf{W}_t and \mathbf{W}_r can be seen as a mapping from the antenna domain onto a beam domain and the entries of the matrix \mathbf{H}_v can be interpreted as the channel gains between the N_t transmit and the N_r receive beams[34], [35]. This model is shown in Fig. 1.

Assuming the channel is time-invariant in the blocks $t \in \{1, \dots, T\}$. Then, $\mathbf{Y} \triangleq [\mathbf{y}_1, \dots, \mathbf{y}_T]$, $\mathbf{S} \triangleq [\mathbf{s}_1, \dots, \mathbf{s}_T]$, and $\mathbf{N} \triangleq [\mathbf{n}_1, \dots, \mathbf{n}_T]$. We rewritten the channel model as

$$\mathbf{Y} = \mathbf{W}_r \mathbf{H}_v \mathbf{W}_t^H \mathbf{S} + \mathbf{N}, \quad (9)$$

where $\mathbf{Y} \in \mathbb{C}^{N_r \times T}$, $\mathbf{S} \in \mathbb{C}^{N_t \times T}$, and $\mathbf{N} \in \mathbb{C}^{N_r \times T}$. We define $\mathbf{X} \triangleq \mathbf{W}_t^H \mathbf{S}$. Then, we recast the (9) as

$$\mathbf{Y} = \mathbf{W}_r \mathbf{H}_v \mathbf{X} + \mathbf{N}. \quad (10)$$

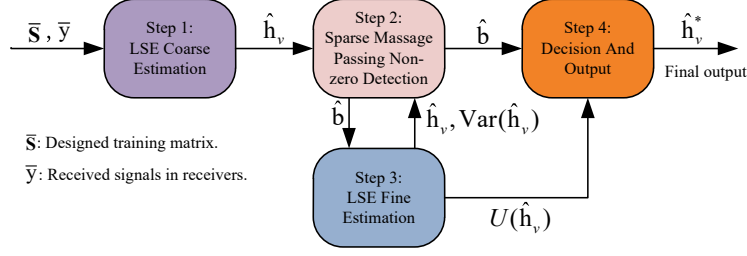


Fig. 2. The processing for the proposed LSE-SMP algorithm, which consists of four phases: Coarse Estimation, Sparse Message Passing Detection, Fine Estimation, and Decision And Output.

Then, we can factorize the (10) [23] as

$$\begin{aligned}
 \text{vec}(\mathbf{Y}) &= \text{vec}(\mathbf{W}_r \mathbf{H}_v \mathbf{X} + \mathbf{N}) \\
 &\stackrel{(a)}{=} (\mathbf{X}^H \otimes \mathbf{W}_r) \text{vec}(\mathbf{H}_v) + \text{vec}(\mathbf{N}),
 \end{aligned} \tag{11}$$

where (a) follows from the equality $\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})\text{vec}(\mathbf{B})$. By defining the $\bar{\mathbf{S}} \triangleq \mathbf{X}^H \otimes \mathbf{W}_r$, we simplify the (11) in a compact fashion:

$$\bar{\mathbf{y}} = \bar{\mathbf{S}} \mathbf{h}_v + \bar{\mathbf{n}}, \tag{12}$$

where $\bar{\mathbf{y}} \in \mathbb{C}^{N_r T \times 1}$, $\bar{\mathbf{S}} \in \mathbb{C}^{N_r T \times N_t N_r}$, $\mathbf{h}_v \in \mathbb{C}^{N_t N_r \times 1}$ and $\bar{\mathbf{n}} \in \mathbb{C}^{N_r T \times 1}$ follows the Gaussian distribution $\mathcal{N}(0, \sigma_n^2 \mathbf{I})$. The mmWave channel estimation problem is simplified to estimate the virtual channel vector \mathbf{h}_v by given the equivalent training matrix $\bar{\mathbf{S}}$ and the observed vector $\bar{\mathbf{y}}$.

III. SPARSE CHANNEL ESTIMATION

In this section, we present an iterative channel estimation algorithm based on the LSE and SMP algorithm as shown in the Fig. 2, which is named LSE-SMP. It consists of four phases: Coarse Estimation, Sparse Message Passing Detection, Fine Estimation, and Decision And Output. Since there is no priori knowledge of \mathbf{h}_v , we initially adopt the LSE method to obtain its coarse estimation. Then, we consider the estimation of non-zero positions in the channel vector \mathbf{h}_v as a detection problem, and propose a SMP algorithm to find these non-zero positions. Again, we apply LSE method by leveraging the estimated non-zero positions to obtain the fine estimation. The step 2 and step 3 will repeat until we obtain a steady estimation. The fourth step is to make a decision based the estimation of step 2 and step 3, and finally output the estimation of the channel vector \mathbf{h}_v .

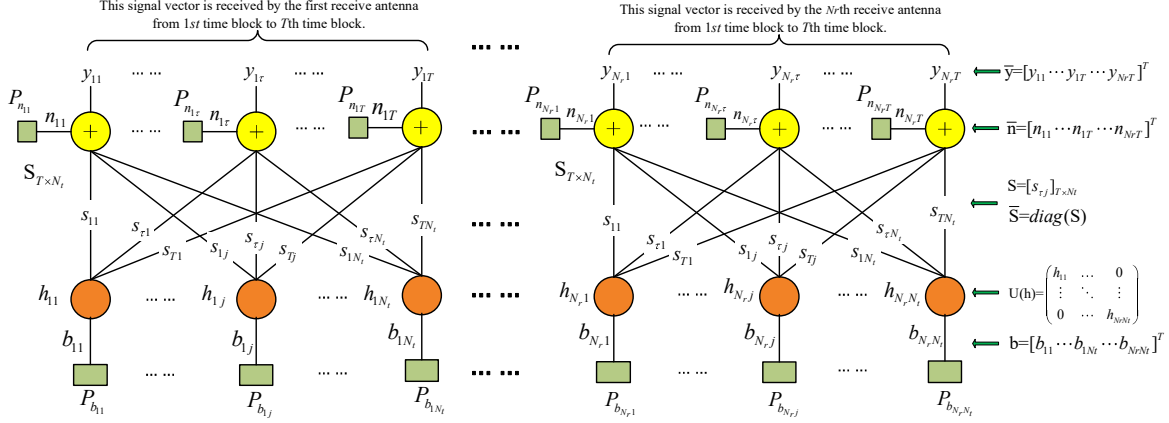


Fig. 3. The factor graph representation for the proposed sparse message passing detection algorithm. This factor graph is plotted on the basis of equations (14)-(18) and factor graph rules. The nodes $(n_{11}, \dots, n_{N_r T})$ and $(h_{11}, \dots, h_{N_r N_t})$ are named the sum and variable nodes respectively.

A. LSE Coarse Estimation

To find the Coarse Estimation $\hat{\mathbf{h}}_v$ based on the observed vector $\bar{\mathbf{y}}$ with a Mean Square Error (MSE) $E\{\|\hat{\mathbf{h}}_v - \mathbf{h}_v\|^2\}$, we can solve the following Least Square (LS) problem,

$$\begin{aligned} \hat{\mathbf{h}}_v &= \arg \min_{\mathbf{h}_v} \|\bar{\mathbf{y}} - \bar{\mathbf{S}}\mathbf{h}_v\|_2^2 \\ &= [\bar{\mathbf{S}}^T \bar{\mathbf{S}}]^{-1} \bar{\mathbf{S}}^T \bar{\mathbf{y}}. \end{aligned} \quad (13)$$

It is noted that it is MVUE in the sense of MSE when the estimator does not have any prior knowledge about neither the sparsity structure of \mathbf{h}_v (i.e., the distribution and location of non-zero entries), nor its degree of sparsity (i.e., L).

B. Sparse Message Passing Algorithm

After we get the Coarse Estimation of \mathbf{h}_v , we propose a fast iterative algorithm to find the positions of non-zero entries. This algorithm is named sparse message passing since it can take full advantage of the channel sparsity and message passing.

1) *Factor Graph Representation of the mmWave Channel:* In order to get better understanding of our proposed algorithm, we show the factor graph representation of the channel vector \mathbf{h}_v in the following. Firstly, we decompose the \mathbf{h}_v into a diagonal coefficient matrix $\mathbf{U}(h_{ij})(i \in \{0, \dots, N_r\}, j \in \{0, \dots, N_t\})$ and a column array \mathbf{b} . The column array $\mathbf{b} = [b_{ij}]_{N_r N_t \times 1}$ is called the position vector, and it represents the positions of non-zero in the virtual channel vector \mathbf{h}_v . The $b_{ij} \in \{1, 0\}$ can be seen as a Bernoulli distribution. Then, the \mathbf{h}_v can be recast as

$$\mathbf{h}_v = \underbrace{\begin{bmatrix} h_{11} & & & 0 \\ & \ddots & & \\ & & h_{1N_t} & \\ & & & \ddots \\ 0 & & & & h_{N_r N_t} \end{bmatrix}}_{=\mathbf{U}(\mathbf{h}_v)} \underbrace{\begin{bmatrix} b_{11} \\ \vdots \\ b_{1N_t} \\ \vdots \\ b_{N_r N_t} \end{bmatrix}}_{=\mathbf{b}}. \quad (14)$$

The equivalent training matrix $\bar{\mathbf{S}}$ can be expressed as the following block diagonal matrix by designing \mathbf{X} and \mathbf{W}_r .

$$\bar{\mathbf{S}} = \text{diag}(\mathbf{S}^T), \quad (15)$$

where $\bar{\mathbf{S}} \in \mathbb{C}^{N_r T \times N_r N_t}$, and \mathbf{S}^T is a training sequence matrix, which will be transmitted by the N_t transmit antennas from the 1st to T th time block. It can be written as

$$\mathbf{S}^T = [s_{\tau j}]_{T \times N_t}, \quad (16)$$

where $s_{\tau j}$ represents a transmitted symbol by the j th ($j \in \{1, \dots, N_t\}$) transmit antenna in the τ th ($\tau \in \{1, \dots, T\}$) time block. Then, we rewrite (12) as

$$\underbrace{\begin{bmatrix} y_{11} & \dots & y_{1T} & \dots & y_{N_r T} \end{bmatrix}^T}_{\bar{\mathbf{y}}} = \bar{\mathbf{S}}\mathbf{h}_v + \bar{\mathbf{n}} \quad (17)$$

$$= \bar{\mathbf{S}}\mathbf{U}(\mathbf{h}_v)\mathbf{b} + \bar{\mathbf{n}}.$$

According to factor graph analysis rules [37], [15], [38], [39], we can plot the factor graph to represent above equations, and it is shown in the Fig. 3. The nodes $(n_{11}, \dots, n_{N_r T})$ and $(h_{11}, \dots, h_{N_r N_t})$ are named the sum and variable nodes respectively. The proposed SMP algorithm is considered for estimating positions of non-zero entries. It is similar to the belief propagation decoding process of the low density parity check code, in which the output message called extrinsic information on each edge is calculated by the messages on the other edges that are connected with the same node [15], [41], [42], [43], [44].

2) *Message Update at Sum Nodes* : To analyze a sum node that is shown in the Fig. 4, we can obtain the received signal by the i th receive antenna at the τ th time block, and it can be expressed as

$$y_{i\tau} = \sum_{j=1}^{N_t} s_{\tau j} h_{ij} b_{ij} + n_{i\tau}. \quad (18)$$

From the assumption of the channel model in the section II, we can know that there are L non-zero entries in the vector of \mathbf{b} . Furthermore, assuming that the b_{ij} is independent and identically

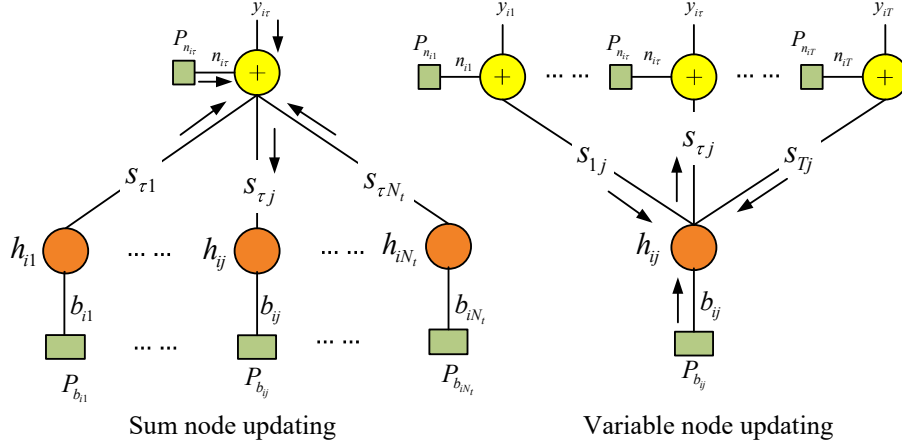


Fig. 4. Messages update at sum nodes and variable nodes. The output message called extrinsic information on each edge is calculated by the messages on the other edges that are connected with the same node. For the Gaussian-Bernoulli sparse signals, the messages passing on each edge are the probabilities of a Bernoulli distribution. The mean and variance of a Gaussian distribution are updated at the sum nodes, and they are used for computing the probability of a Bernoulli distribution.

distributed (i.i.d.). Then, we can know the initial probability of the Bernoulli distribution, which can be denoted as

$$\begin{cases} p_0(b_{ij} = 1) = \frac{L}{N_r N_t}, \\ p_0(b_{ij} = 0) = 1 - \frac{L}{N_r N_t}. \end{cases} \quad (19)$$

$$\quad (20)$$

When the N_t goes very large, we can know that the term $\sum_{j=1}^{N_t} s_{\tau j} h_{ij} b_{ij}$ can be approximated as the Gaussian distribution [39], [44], [33], [45] according to the law of large numbers. When we compute the probability of $p(b_{ij} = 1)$ from the $i\tau$ sum node to the ij variable node, we consider the messages from the other variable nodes im ($m \neq j$ and $m \in \{1, 2, \dots, N_t\}$) to the sum node $i\tau$ as the **equivalent Gaussian noise** $n_{i\tau}^*$. This can be expressed as

$$y_{i\tau} = \underbrace{s_{\tau j} h_{ij} p(b_{ij} = 1)}_{\text{Desired}} + \underbrace{\sum_{m \neq j}^{N_t} s_{\tau m} h_{im} p(b_{im} = 1)}_{\text{Equivalent Gaussian noise: } n_{i\tau}^*} + n_{i\tau}. \quad (21)$$

Furthermore, we can compute the mean value and variance of the equivalent Gaussian noise.

These messages update at the sum nodes are given by

$$\begin{aligned}
v_{i\tau \rightarrow ij}^s(k) &= Var \left\{ n_{i\tau}^* | \mathbf{s}_\tau, \hat{\mathbf{h}}_i(k), \mathbf{p}_i^v(k), \hat{\mathbf{v}}_{h_i}(k) \right\} \\
&= E \left\{ n_{i\tau}^{*2} | \mathbf{s}_\tau, \hat{\mathbf{h}}_i(k), \mathbf{p}_i^v(k), \hat{\mathbf{v}}_{h_i}(k) \right\} - E^2 \left\{ n_{i\tau}^* | \mathbf{s}_\tau, \hat{\mathbf{h}}_i(k), \mathbf{p}_i^v(k) \right\} \\
&= \sum_{m \neq j} s_{\tau m}^2 E \left\{ \hat{h}_{im}^2(k) \right\} E \left\{ p_{im \rightarrow i\tau}^{v2}(k) \right\} - s_{\tau m}^2 \hat{h}_{im}^2(k) p_{im \rightarrow i\tau}^{v2}(k) \\
&= \sum_{m \neq j} s_{\tau m}^2 \left((v_{h_{im}}(k) + \hat{h}_{im}^2(k)) p_{im \rightarrow i\tau}^v(k) \right) - s_{\tau m}^2 \hat{h}_{im}^2(k) p_{im \rightarrow i\tau}^{v2}(k) \\
&= \sum_{m \neq j} s_{\tau m}^2 p_{im \rightarrow i\tau}^v(k) \left(v_{h_{im}}(k) + \hat{h}_{im}^2(k) (1 - p_{im \rightarrow i\tau}^v(k)) \right) + \sigma_n^2.
\end{aligned} \tag{22}$$

$$\begin{aligned}
e_{i\tau \rightarrow ij}^s(k) &= E \left\{ n_{i\tau}^* | \mathbf{s}_\tau, \hat{\mathbf{h}}_i(k), \mathbf{p}_i^v(k) \right\} \\
&= \sum_{m \neq j} s_{\tau m} \hat{h}_{im}(k) p_{im \rightarrow i\tau}^v(k),
\end{aligned} \tag{23}$$

where $j, m \in \{1, 2, \dots, N_t\}$, $i \in \{1, 2, \dots, N_r\}$, $\tau \in \{1, 2, \dots, T\}$, and σ_n^2 is the variance of the Gaussian noise, and $\mathbf{s}_\tau = [s_{\tau 1}, \dots, s_{\tau N_t}]^T$, $\hat{\mathbf{h}}_i = [\hat{h}_{i1}, \dots, \hat{h}_{iN_t}]^T$, $\mathbf{p}_i^v = [p_{i1 \rightarrow i\tau}^v, \dots, p_{iN_t \rightarrow i\tau}^v]^T$ and $\hat{\mathbf{v}}_{h_i} = [v_{h_{i1}}, \dots, v_{h_{iN_t}}]^T$. In addition, $e_{i\tau \rightarrow ij}^s(k)$ and $v_{i\tau \rightarrow ij}^s(k)$ denote the mean and variance of the equivalent Gaussian noise $n_{i\tau}^*$ when the message $s_{\tau j} h_{ij} p(b_{ij} = 1)$ passes from the $i\tau$ sum node to the ij variable node at the k th iteration. Similarly, $p_{im \rightarrow i\tau}^v(k)$ denotes the probability message of $p(b_{im} = 1)$ passing from the im variable node to the $i\tau$ sum node at the k th iteration. \hat{h}_{im} and $v_{h_{im}}$ denote that the mean and variance of h_{im} are estimated in the fine phase.

Once we obtain the mean and variance of the equivalent Gaussian noise, we can compute the statistical probability of $b_{ij} = 1$ in according to $(y_{i\tau} - s_{\tau j} h_{ij} p(b_{ij} = 1)) \sim \mathcal{N}(e_{i\tau \rightarrow ij}^s(k), v_{i\tau \rightarrow ij}^s(k))$. Firstly, we define the Gaussian probability density function as

$$f(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}}. \tag{24}$$

Then, we can give the probability message of $b_{ij} = 1$ passing from the $i\tau$ sum node to the ij

variable node as follows

$$\begin{aligned}
p_{i\tau \rightarrow ij}^s(k) &= \frac{1}{1 + \frac{P(b_{ij}=0|y_{i\tau}, s_{\tau j}(k), \hat{h}_{ij}(k), v_{h_{ij}}(k), e_{i\tau \rightarrow ij}^s(k), v_{i\tau \rightarrow ij}^s(k))}{P(b_{ij}=1|y_{i\tau}, s_{\tau j}(k), \hat{h}_{ij}(k), v_{h_{ij}}(k), e_{i\tau \rightarrow ij}^s(k), v_{i\tau \rightarrow ij}^s(k))}} \\
&= \frac{P(y_{i\tau} = s_{\tau j} h_{ij} + n_{i\tau}^*)}{P(y_{i\tau} = s_{\tau j} h_{ij} + n_{i\tau}^*) + P(y_{i\tau} = n_{i\tau}^*)} \\
&= \frac{f(y_{i\tau} | e_{i\tau \rightarrow ij}^s(k) + s_{\tau j} \hat{h}_{ij}(k), v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k))}{f(y_{i\tau} | e_{i\tau \rightarrow ij}^s(k) + s_{\tau j} \hat{h}_{ij}(k), v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k)) + f(y_{i\tau} | e_{i\tau \rightarrow ij}^s(k), v_{i\tau \rightarrow ij}^s(k))} \\
&= \frac{1}{1 + \sqrt{\frac{v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k)}{v_{i\tau \rightarrow ij}^s(k)}} \exp \left(\frac{(y_{i\tau} - e_{i\tau \rightarrow ij}^s(k) - s_{\tau j} \hat{h}_{ij}(k))^2}{2(v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k))} - \frac{(y_{i\tau} - e_{i\tau \rightarrow ij}^s(k))^2}{2v_{i\tau \rightarrow ij}^s(k)} \right)}.
\end{aligned} \tag{25}$$

3) *Message Update at Variable Nodes* : In terms of the message update at variable nodes, we consider variable nodes as a broadcast process [46], [47] and the message update at the variable node is given by

$$p_{ij \rightarrow i\tau}^v(k+1) = \frac{\prod_{t \neq \tau} p_{it \rightarrow ij}^s(k) \cdot p_0(b_{ij} = 1)}{\prod_{t \neq \tau} p_{it \rightarrow ij}^s(k) \cdot p_0(b_{ij} = 1) + \prod_{t \neq \tau} (1 - p_{it \rightarrow ij}^s(k)) \cdot p_0(b_{ij} = 0)}. \tag{26}$$

where $t, \tau \in \{1, 2, \dots, T\}$, and $p_{it \rightarrow ij}^s(k+1)$ denotes the probability message of $b_{ij} = 1$ passing from the it sum node to the ij variable node at the $(k+1)$ th iteration. Furthermore, we can obtain the estimation of the Bernoulli variable b_{ij} at the $(k+1)$ th iteration as

$$\hat{b}_{ij}(k+1) = \frac{\prod_{t=1}^T p_{it \rightarrow ij}^s(k) \cdot p_0(b_{ij} = 1)}{\prod_{t=1}^T p_{it \rightarrow ij}^s(k) \cdot p_0(b_{ij} = 1) + \prod_{t=1}^T (1 - p_{it \rightarrow ij}^s(k)) \cdot p_0(b_{ij} = 0)}. \tag{27}$$

It should be pointed out that $p_{it \rightarrow ij}^s(k+1)$ is the extrinsic information and will be used to update messages of the sum nodes in the next iteration. On the other hand, $\hat{b}_{ij}(k+1)$ is updated based on the full information coming from all the sum nodes, and it will be used in the Fine Estimation phase for the estimation of \mathbf{h}_v .

C. LSE Fine Estimation

Once the positions of the non-zero entries have been estimated, the next step is to estimate the value of the coefficient matrix $\mathbf{U}(\mathbf{h}_v)$. For the problem, we propose a novel strategy based on the LSE method. This strategy is to swap the position of h_{ij} and b_{ij} in the (17), so that we can get an accurate estimation by leveraging the sparsity of \mathbf{b} . Rewriting the (17) as

$$\bar{\mathbf{y}} = \bar{\mathbf{S}}\mathbf{U}(\mathbf{b})\mathbf{h}_v + \bar{\mathbf{n}}, \tag{28}$$

where $\mathbf{h}_v \in \mathbb{C}^{N_t N_r \times 1}$. Similar with the Coarse Estimation, the estimation of \mathbf{h}_v can be obtained by solving the following LS problem

$$\hat{\mathbf{h}}_v = \arg \min_{\mathbf{h}_v} \{\|\bar{\mathbf{y}} - \bar{\mathbf{S}}\mathbf{U}(\hat{\mathbf{b}})\mathbf{h}_v\|_2^2\}, \quad (29)$$

where $\hat{\mathbf{b}}$ is the vector estimated by the SMP algorithm. Solving the above expression, we get the following estimator for \mathbf{h}_v as

$$\begin{cases} \hat{\mathbf{h}}_v(k) = \hat{\mathbf{Q}}^\dagger(k) \left(\bar{\mathbf{S}}\mathbf{U}(\hat{\mathbf{b}}(k)) \right)^T \bar{\mathbf{y}}, \\ \hat{\mathbf{v}}_h(k) = \sigma_n^2 \left(\left(\bar{\mathbf{S}}\mathbf{U}(\hat{\mathbf{b}}(k)) \right)^T \bar{\mathbf{S}}\mathbf{U}(\hat{\mathbf{b}}(k)) \right)^\dagger, \end{cases} \quad (30)$$

$$\quad (31)$$

where $\hat{\mathbf{Q}}(k) = \mathbf{U}(\hat{\mathbf{b}}(k))\bar{\mathbf{S}}^T\bar{\mathbf{S}}\mathbf{U}(\hat{\mathbf{b}}(k))$, $\hat{\mathbf{v}}_h = [v_{h_{im}}]_{N_r N_t \times 1}$ ($i \in \{1, 2, \dots, N_r\}$, $m \in \{1, 2, \dots, N_t\}$). $\hat{\mathbf{h}}_v(k)$ and $\hat{\mathbf{v}}_h(k)$ denotes the estimated value and variance of \mathbf{h}_v at k th iteration. After we obtain $\hat{\mathbf{h}}_v(k)$ and $\hat{\mathbf{v}}_h(k)$, these values will replace the $\hat{\mathbf{h}}_v(k)$ and $\hat{\mathbf{v}}_h(k)$ for calculating the mean and variance of the equivalent Gaussian noise in the next iteration.

D. Decision And Output of LSE-SMP

When the MSE of the LSE-SMP meets the requirement or the number of iterations reaches the limit, we output the final estimation of channel vector \mathbf{h}_v as

$$\begin{cases} \hat{\mathbf{h}}_v = \hat{\mathbf{Q}}^\dagger(k) \left(\bar{\mathbf{S}}\mathbf{U}(\hat{\mathbf{b}}(k)) \right)^T \bar{\mathbf{y}}, \\ \hat{\mathbf{h}}_v^* = \mathbf{U}(\hat{\mathbf{h}}_v)\hat{\mathbf{b}}, \end{cases} \quad (32)$$

$$\quad (33)$$

where $\hat{\mathbf{b}} = [\hat{b}_{ij}]_{N_r N_t \times 1}$, $i \in \{1, 2, \dots, N_r\}$ and $j \in \{1, 2, \dots, N_t\}$. It should be pointed out that the final output is based on the SMP and LSE Fine Estimation.

E. LLRs of LSE-SMP

From (26) and (27), we notice that the messages update for the variable nodes are easy to overflow in the simulation due to the multiplications of a large number of probabilities. Therefore, we use Log-Likelihood Ratios (LLRs) scheme [44], [48] to replace the computation of the non-zero probabilities during the message update process. This not only can prevent the overflow, but also can reduce the complexity of computation. The LLRs scheme can be written as follows

$$\begin{cases} l_{ij \rightarrow i\tau}^s(k) = \log \frac{p_{ij \rightarrow i\tau}^s(k)}{1 - p_{ij \rightarrow i\tau}^s(k)}, \end{cases} \quad (34)$$

$$\begin{cases} l_{i\tau \rightarrow ij}^v(k) = \log \frac{p_{i\tau \rightarrow ij}^v(k)}{1 - p_{i\tau \rightarrow ij}^v(k)}, \end{cases} \quad (35)$$

$$\begin{cases} l_0 = \log \frac{P_0(b_{ij} = 1)}{1 - P_0(b_{ij} = 1)}, \end{cases} \quad (36)$$

for any i ($i \in \{1, \dots, N_r\}$), j ($j \in \{1, \dots, N_t\}$), τ ($\tau \in \{1, \dots, T\}$) and k (k denotes the number of iterations). $l_{ij \rightarrow i\tau}^s(k)$ denotes the LLRs of the probability message of $b_{ij} = 1$ passing from the ij variable node to $i\tau$ sum node. Similarly, $l_{i\tau \rightarrow ij}^v(k)$ denotes the LLRs of the probability message of $b_{ij} = 1$ passing from the $i\tau$ sum node to ij variable node. Then, the LSE-SMP algorithm is updated by LLRs as follows.

1) *Message Update at Sum Nodes* : Based on (23), (22) and (25), the LLRs of Bernoulli-Gaussian messages updating at sum nodes are rewritten as

$$e_{i\tau \rightarrow ij}^s(k) = \sum_{m \neq j} s_{\tau m} \hat{h}_{im}(k) p_{im \rightarrow i\tau}^v(k), \quad (37)$$

$$v_{i\tau \rightarrow ij}^s(k) = \sum_{m \neq j} s_{\tau m}^2 p_{im \rightarrow i\tau}^v(k) \left(v_{h_{im}}(k) + \hat{h}_{im}^2(k) (1 - p_{im \rightarrow i\tau}^v(k)) \right) + \sigma_n^2, \quad (38)$$

$$l_{i\tau \rightarrow ij}^s(k) = -\log \left(\sqrt{\frac{v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k)}{v_{i\tau \rightarrow ij}^s(k)}} \right) - \frac{(y_{i\tau} - e_{i\tau \rightarrow ij}^s(k) - s_{\tau j} \hat{h}_{ij}(k))^2}{2(v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k))} + \frac{(y_{i\tau} - e_{i\tau \rightarrow ij}^s(k))^2}{2v_{i\tau \rightarrow ij}^s(k)}, \quad (39)$$

where $p_{im \rightarrow i\tau}^v = 1/(1 + e^{-l_{im \rightarrow i\tau}^v})$.

2) *Message Update at Variable Nodes* : Based on (26) and (27), the LLRs of messages updating at variable nodes are rewritten as

$$l_{ij \rightarrow i\tau}^v(k+1) = l_0 + \sum_{t \neq \tau}^T l_{it \rightarrow ij}^s(k), \quad (40)$$

$$\hat{b}_{ij}(k+1) = l_0 + \sum_{t=1}^T l_{it \rightarrow ij}^s(k). \quad (41)$$

3) *Decision and Output of LSE-SMP*: Similarly, we can obtain the final output LLRs of the LSE-SMP as follows

$$\hat{\mathbf{h}}_v = \hat{\mathbf{Q}}^\dagger(k) \left(\bar{\mathbf{S}} \mathbf{U} \left(\hat{\mathbf{b}}(k) \right) \right)^T \bar{\mathbf{y}}, \quad (42)$$

$$\hat{\mathbf{h}}_v^* = \mathbf{U}(\hat{\mathbf{h}}_v) \hat{\mathbf{b}}. \quad (43)$$

From (37)-(43), it should be noticed that the LSE-SMP algorithm in the LLRs form will be concise, which can reduce the complexity of computation and prevent the overflow of multiplications of a large number of probabilities, since LLRs transforms the multiplication operations into the addition operations. In addition, we also can see that it will be convenient to analyze and predict the performance of the system in the following section IV.

F. LSE-SMP in the Matrix Form

From the (15), (16) and (17), we know that $\bar{\mathbf{S}}$ is a block diagonal matrix. In order to reduce the complexity of matrix inversions and multiplications, we can split the high dimension diagonal matrix and block diagonal matrix into some low dimension matrixs, since the transmit and receive antennas are independent each other as we mentioned before. As the definitions before, we denote $i \in \{1, 2, \dots, N_r\}, j \in \{1, 2, \dots, N_t\}$ and $\tau \in \{1, 2, \dots, T\}$, and k denotes the k th iteration in the following definitions. Then, we give some definitions as follows: $\mathbf{U}^s(k) = \text{diag}[\mathbf{U}_i^s(k)]_{N_r T \times N_r N_t}$, $\mathbf{U}_i^s(k) = [e_{i\tau \rightarrow ij}^s(k)]_{T \times N_t}$; $\mathbf{V}^s(k) = \text{diag}[\mathbf{V}_i^s(k)]_{N_r T \times N_r N_t}$, $\mathbf{V}_i^s(k) = [v_{i\tau \rightarrow ij}^s(k)]_{T \times N_t}$; $\mathbf{P}^s(k) = \text{diag}[\mathbf{P}_i^s(k)]_{N_r T \times N_r N_t}$, $\mathbf{P}_i^s(k) = [p_{i\tau \rightarrow ij}^s(k)]_{T \times N_t}$; $\mathbf{L}^s(k) = \text{diag}[\mathbf{L}_i^s(k)]_{N_r T \times N_r N_t}$, $\mathbf{L}_i^s(k) = [l_{i\tau \rightarrow ij}^s(k)]_{T \times N_t}$; $\mathbf{P}^v(k) = \text{diag}[\mathbf{P}_i^v(k)]_{N_r N_t \times N_r T}$, $\mathbf{P}_i^v(k) = [p_{ij \rightarrow i\tau}^v(k)]_{N_t \times T}$; $\mathbf{L}^v(k) = \text{diag}[\mathbf{L}_i^v(k)]_{N_r N_t \times N_r T}$, $\mathbf{L}_i^v(k) = [l_{ij \rightarrow i\tau}^v(k)]_{N_t \times T}$; $\mathbf{U}(\hat{\mathbf{h}}_i)(k) = \text{diag}[\mathbf{U}(\hat{\mathbf{h}}_i)(k)]_{N_r N_t \times N_r N_t}$, $\mathbf{U}(\hat{\mathbf{h}}_i)(k) = [\hat{h}_{ij}(k)]_{N_t \times N_t}$; $\hat{\mathbf{h}}_v(k) = [\hat{\mathbf{h}}_i(k)]_{N_r N_t \times 1}^H$, $\hat{\mathbf{h}}_i(k) = [\hat{h}_{ij}(k)]_{N_t \times 1}^H$; $\hat{\mathbf{V}}_h(k) = \text{diag}[\hat{\mathbf{V}}_{h_i}(k)]_{N_r N_t \times N_r N_t}^H$, $\hat{\mathbf{V}}_{h_i}(k) = [v_{h_{ij}}(k)]_{N_t \times N_t}$; $\bar{\mathbf{y}} = [\mathbf{y}_i]_{N_r N_t \times 1}^H$, $\mathbf{y}_i = [y_{it}]_{T \times 1}^H$; $\mathbf{U}(\hat{\mathbf{b}}_i(k)) = \text{diag}[\mathbf{U}(\hat{\mathbf{b}}_i(k))]_{N_r N_t \times N_r N_t}^T$, $\mathbf{U}(\hat{\mathbf{b}}_i(k)) = [\hat{b}_{ij}(k)]_{N_t \times N_t}^T$, $\hat{\mathbf{Q}}_i(k) = \mathbf{U}(\hat{\mathbf{b}}_i(k))\mathbf{S}\mathbf{S}^T\mathbf{U}(\hat{\mathbf{b}}_i(k))$. Let $\mathbf{A}_{N_r \times N_t} * \mathbf{B}_{N_r \times N_t} = [a_{ij}b_{ij}]_{N_r \times N_t}$, $\mathbf{A}_{N_r \times N_t}^{(2)} = [a_{ij}^2]_{N_r \times N_t}$, $\mathbf{1}_{N_r \times N_t} = [1]_{N_r \times N_t}$, and $\mathbf{C}_{N_r \times N_r} = \mathbf{A}_{N_r \times N_t} \cdot \mathbf{B}_{N_t \times N_r}$ is the matrix product of \mathbf{A} and \mathbf{B} . The message update at sum node is rewritten in matrix form as

$$\begin{bmatrix} \mathbf{U}_i^s(k) \\ \mathbf{V}_i^s(k) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{U}}^{*H}(k) \cdot \mathbf{1}_{N_t \times 1} \\ \sigma_n^2 \cdot \mathbf{1}_{T \times 1} + \{\tilde{\mathbf{V}}^{*H}(k) \cdot \mathbf{1}_{N_t \times 1}\} \end{bmatrix} \cdot \mathbf{1}_{1 \times N_t} - \begin{bmatrix} \tilde{\mathbf{U}}^{*H}(k) \\ \tilde{\mathbf{V}}^{*H}(k) \end{bmatrix}, \quad (44)$$

where $\tilde{\mathbf{V}}^*(k) = \mathbf{S}^{(2)} * (\hat{\mathbf{V}}_{h_i}(k) \cdot \mathbf{P}_i^v(k) + \mathbf{U}(\hat{\mathbf{h}}_i)^{(2)}(k) \cdot \mathbf{P}_i^v(k) \cdot (\mathbf{1}_{N_t \times T} - \mathbf{P}_i^v(k)))$, $\tilde{\mathbf{U}}^*(k) = \mathbf{S} \cdot \mathbf{U}(\hat{\mathbf{h}}_i)(k) \cdot \mathbf{P}_i^v(k)$, and $\mathbf{P}_i^v(k) = (\mathbf{1}_{N_t \times T} + e^{-\mathbf{L}_i^v(k)})^{-1}$. Before the first iteration, we can obtain the initial estimation of $\hat{\mathbf{h}}_i(0) = [\mathbf{S}\mathbf{S}^T]^{-1}\mathbf{S}\mathbf{y}_i$, $\hat{\mathbf{V}}_{h_i}(0) = \sigma_n^2[\mathbf{S}\mathbf{S}^T]^{-1}$, and $\mathbf{L}_i^s(0) = \log \frac{L}{N_r N_t - L} \cdot \mathbf{1}_{T \times N_t}$.

$$[\mathbf{L}_i^s(k)] = \left[\frac{1}{2} \log \frac{\mathbf{V}_i^s(k) + \hat{\mathbf{V}}_{h_i}(k) \cdot \mathbf{S}^{(2)}}{\mathbf{V}_i^s(k)} + \frac{(\bar{\mathbf{y}}_i \cdot \mathbf{1}_{1 \times N_t} - \mathbf{U}_i^s(k))^2}{2\mathbf{V}_i^s(k)} - \frac{(\bar{\mathbf{y}}_i \cdot \mathbf{1}_{1 \times N_t} - \mathbf{U}_i^s(k) - \mathbf{S}^T \cdot \mathbf{U}(\hat{\mathbf{h}}_i)(k))^2}{2(\mathbf{V}_i^s(k) + \hat{\mathbf{V}}_{h_i}(k) \cdot \mathbf{S}^{(2)})} \right], \quad (45)$$

The message update at variable node is rewritten as:

$$\begin{bmatrix} \mathbf{L}_i^v(k+1) \\ \hat{\mathbf{b}}_i^v(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{1}_{N_t \times 1} \cdot [\mathbf{1}_{1 \times N_t} \cdot (\mathbf{L}_i^s(k))^H] + (\mathbf{L}_i^s(0))^H - (\mathbf{L}_i^s(k))^H \\ \mathbf{1}_{N_t \times 1} \cdot [\mathbf{1}_{1 \times N_t} \cdot (\mathbf{L}_i^s(k))^H] + (\mathbf{L}_i^s(0))^H \end{bmatrix}, \quad (46)$$

The LSE Fine Estimation is rewritten as:

$$\begin{bmatrix} \hat{\mathbf{h}}_i(k) \\ \hat{\mathbf{V}}_{h_i}(k) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{Q}}_i^\dagger(k) \left(\mathbf{S}^T \mathbf{U}(\hat{\mathbf{b}}_i(k)) \right)^T \bar{\mathbf{y}}_i \\ \sigma_n^2 \left(\hat{\mathbf{Q}}_i^\dagger(k) \right)^\dagger \end{bmatrix}, \quad (47)$$

when $|\mathbf{U}(\hat{\mathbf{h}}_v)(k+1) - \mathbf{U}(\hat{\mathbf{h}}_v)(k)| < \epsilon$ and $|\mathbf{L}^v(k+1) - \mathbf{L}^v(k)| < \epsilon$, where $\epsilon > 0$ is a sufficiently small precision parameter, or when the evaluation has passed sufficient number of iterations, we output $\hat{\mathbf{h}}_v$ and $\hat{\mathbf{h}}_v^*$ as follows

$$\begin{bmatrix} \hat{\mathbf{h}}_v \\ \hat{\mathbf{h}}_v^* \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{Q}}^\dagger (\bar{\mathbf{S}} \mathbf{U}(\hat{\mathbf{b}}))^T \bar{\mathbf{y}} \\ \mathbf{U}(\hat{\mathbf{h}}_v) \hat{\mathbf{b}} \end{bmatrix}. \quad (48)$$

The algorithm 1 shows the detailed process of the LSE-SMP.

Algorithm 1 LSE-SMP Algorithm

- 1: **Input:** \mathbf{S} , $L, T, N_r, N_t, \sigma_n^2, \epsilon > 0, N_{ite}$ and calculate $\mathbf{S}^{(2)}, (\mathbf{S}^T \mathbf{S})^{-1}$,
 - 2: **Initialized Coarse LSE Estimation:** $\hat{\mathbf{h}}_i(0) = [\mathbf{S} \mathbf{S}^T]^{-1} \mathbf{S} \mathbf{y}_i$, $\hat{\mathbf{V}}_{h_i}(0) = \sigma_n^2 [\mathbf{S} \mathbf{S}^T]^{-1}$, and $\mathbf{L}_i^s(0) = \log \frac{L}{N_r N_t - L} \cdot \mathbf{1}_{T \times N_t}$,
 - 3: **for** $i = 1 : N_r$
 - 4: **Do**
 - 5: $\tilde{\mathbf{V}}^*(k) = \mathbf{S}^{(2)} \cdot * \left(\hat{\mathbf{V}}_{h_i}(k) \cdot \mathbf{P}_i^v(k) + \mathbf{U}(\hat{\mathbf{h}}_i)^{(2)}(k) \cdot \mathbf{P}_i^v(k) \cdot * (\mathbf{1}_{N_t \times T} - \mathbf{P}_i^v(k)) \right)$,
 - 6: $\tilde{\mathbf{U}}^*(k) = \mathbf{S} \cdot \mathbf{U}(\hat{\mathbf{h}}_i)(k) \cdot * \mathbf{P}_i^v(k)$, and $\mathbf{P}_i^v(k) = (\mathbf{1}_{N_t \times T} + e^{-\mathbf{L}_i^v(k)})^{-1}$,
 - 7: $\begin{bmatrix} \mathbf{U}_i^s(k) \\ \mathbf{V}_i^s(k) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{U}}^{*H}(k) \cdot \mathbf{1}_{N_t \times 1} \\ \sigma_n^2 \cdot \mathbf{1}_{T \times 1} + \{ \tilde{\mathbf{V}}^{*H}(k) \cdot \mathbf{1}_{N_t \times 1} \} \end{bmatrix} \cdot \mathbf{1}_{1 \times N_t} - \begin{bmatrix} \tilde{\mathbf{U}}^{*H}(k) \\ \tilde{\mathbf{V}}^{*H}(k) \end{bmatrix}$,
 - 8: $\left[\mathbf{L}_i^s(k) \right] = \left[\frac{1}{2} \log \frac{\mathbf{V}_i^s(k) + \hat{\mathbf{V}}_{h_i}(k) \cdot \mathbf{S}^{(2)}}{\mathbf{V}_i^s(k)} + \frac{(\bar{\mathbf{y}}_i \cdot \mathbf{1}_{1 \times N_t} - \mathbf{U}_i^s(k))^2}{2 \mathbf{V}_i^s(k)} - \frac{(\bar{\mathbf{y}}_i \cdot \mathbf{1}_{1 \times N_t} - \mathbf{U}_i^s(k) - \mathbf{S}^T \cdot \mathbf{U}(\hat{\mathbf{h}}_i)(k))^2}{2 (\mathbf{V}_i^s(k) + \hat{\mathbf{V}}_{h_i}(k) \cdot \mathbf{S}^{(2)})} \right]$,
 - 9: $\begin{bmatrix} \mathbf{L}_i^v(k+1) \\ \hat{\mathbf{b}}_i^v(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{1}_{N_t \times 1} \cdot [\mathbf{1}_{1 \times N_t} \cdot (\mathbf{L}_i^s(k))^H] + (\mathbf{L}_i^s(0))^H - (\mathbf{L}_i^s(k))^H \\ \mathbf{1}_{N_t \times 1} \cdot [\mathbf{1}_{1 \times N_t} \cdot (\mathbf{L}_i^s(k))^H] + (\mathbf{L}_i^s(0))^H \end{bmatrix}$,
 - 10: $\begin{bmatrix} \hat{\mathbf{h}}_i(k) \\ \hat{\mathbf{V}}_{h_i}(k) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{Q}}_i^\dagger(k) \left(\mathbf{S}^T \mathbf{U}(\hat{\mathbf{b}}_i(k)) \right)^T \bar{\mathbf{y}}_i \\ \sigma_n^2 \left(\hat{\mathbf{Q}}_i^\dagger(k) \right)^\dagger \end{bmatrix}$,
 - 11: $k = k + 1$.
 - 12: **While** $((|\mathbf{U}(\hat{\mathbf{h}}_v)(k+1) - \mathbf{U}(\hat{\mathbf{h}}_v)(k)| < \epsilon \ \& \ |\mathbf{L}^v(k+1) - \mathbf{L}^v(k)| < \epsilon) \ \text{or} \ k \leq N_{ite})$
 - 13: **end for**
 - 14: $\begin{bmatrix} \hat{\mathbf{h}}_v \\ \hat{\mathbf{h}}_v^* \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{Q}}^\dagger (\bar{\mathbf{S}} \mathbf{U}(\hat{\mathbf{b}}))^T \bar{\mathbf{y}} \\ \mathbf{U}(\hat{\mathbf{h}}_v) \hat{\mathbf{b}} \end{bmatrix}$,
 - 15: **Output:** $\hat{\mathbf{h}}_v$ and $\hat{\mathbf{h}}_v^*$.
-

IV. PERFORMANCE ANALYSIS

A. Cramer Rao Low Bound Of LSE-SMP

In this section, we give the analysis of Cramer-Rao Low Bound and show that our proposed LSE-SMP algorithm is unbiased. Firstly, we consider the case that the channel vector \mathbf{h}_v is a

deterministic and non-sparse. From [49], [50] and the signal model in (13), we can yield the CRLB of the conventional LSE as

$$\text{CRLB}_{\text{LSE}} \geq \mathbf{C}_{\text{LSE}} = \sigma_n^2 (\bar{\mathbf{S}}^T \bar{\mathbf{S}})^{-1}, \quad (49)$$

where \mathbf{C}_{LSE} is the covariance matrix of $\hat{\mathbf{h}}_v$ for the LSE estimation. Note that LSE is the Minimum Variance Unbiased Estimator, and the detailed proof can be found in [49], [50]. This means that the LSE estimator is the optimal estimator if \mathbf{h}_v is a deterministic and non-sparse vector. Compared with the non-sparse case, the sparse case is slightly more complex. Since we are interested in the lower bound for the estimation accuracy, it is very difficult to analyze our estimator directly. Recalling the in (14), we know that \mathbf{h}_v can be decomposed into the two parts ($\mathbf{U}(\mathbf{h}_v)$ and \mathbf{b}). Therefore, we can take the alternating minimization method [51], [52], [53] to analyze these two parts independently. For example, when we analyze the estimated performance of $\mathbf{U}(\mathbf{h}_v)$, we keep the \mathbf{b} invariant. Therefore, we have the assumption that we have perfect knowledge of the non-zero positions, i.e., $\hat{\mathbf{b}} = \mathbf{b}$. Then, we can obtain the following theorem.

Theorem 1: For the deterministic and sparse channel with the assumption that we have the knowledge of the non-zero positions of the channel vector \mathbf{h}_v , the proposed LSE-SMP is the MVUE, and can achieve the CRLB. The CRLB is given by

$$\text{CRLB}_{\text{LSE-SMP}} \geq \sigma_n^2 \left((\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{S}}\mathbf{U}(\mathbf{b}) \right)^\dagger. \quad (50)$$

Proof: The first step is to verify that the our proposed estimator is unbiased. Recalling the signal model in (28) and the definition of the unbiased estimator, we get

$$\begin{aligned} E(\hat{\mathbf{h}}_v) &= E \left\{ \left((\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{S}}\mathbf{U}(\mathbf{b}) \right)^\dagger (\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{y}} \right\} \\ &= E \left\{ \left((\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{S}}\mathbf{U}(\mathbf{b}) \right)^\dagger (\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T (\bar{\mathbf{S}}\mathbf{U}(\mathbf{b})\mathbf{h}_v + \bar{\mathbf{n}}) \right\} \\ &= \mathbf{U}(\mathbf{b})\mathbf{h}_v = \mathbf{h}_v. \end{aligned} \quad (51)$$

So, it is a unbiased estimator. The next step is to compute its CRLB and verify that our proposed LSE-SMP algorithm can achieve the CRLB. As previously mentioned, the channel \mathbf{h}_v is a deterministic vector, and the white Gaussian noise follows the $\mathcal{N}(0, \sigma_n^2 \mathbf{I})$. Recalling the signal model in (12), we can get

$$\bar{\mathbf{y}} \sim f(\bar{\mathbf{y}}|\mathbf{h}_v) = \mathcal{N}(\bar{\mathbf{S}}\mathbf{h}_v, \sigma_n^2 \mathbf{I}), \quad (52)$$

where $f(\bar{\mathbf{y}}|\mathbf{h}_v)$ is the probability density function of $\bar{\mathbf{y}}$ under the condition of known \mathbf{h}_v . Then, we can compute the $\frac{\partial \ln f(\bar{\mathbf{y}}|\mathbf{h}_v)}{\partial \mathbf{h}_v}$,

$$\frac{\partial \ln f(\bar{\mathbf{y}}|\mathbf{h}_v)}{\partial \mathbf{h}_v} = \frac{1}{\sigma_n^2} \left[(\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{y}} - (\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{S}}\mathbf{U}(\mathbf{b})\mathbf{h}_v \right]. \quad (53)$$

Then, we obtain the following expression for the Fisher Information Matrix (FIM):

$$I(\mathbf{h}_v) = -E \left\{ \frac{\partial^2 \ln f(\bar{\mathbf{y}}|\mathbf{h}_v)}{\partial \mathbf{h}_v^2} \right\} = \frac{1}{\sigma_n^2} (\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{S}}\mathbf{U}(\mathbf{b}). \quad (54)$$

We note that $\bar{\mathbf{S}}\mathbf{U}(\mathbf{b})$ has the rank no larger than L due to the multiplication of $\bar{\mathbf{S}}$ by $\mathbf{U}(\mathbf{b})$. The matrices $\bar{\mathbf{S}}\mathbf{U}(\mathbf{b})$ and $(\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T$ have some all zero columns (and rows), so it is singular. For this type singular matrix, it need to meet a constraint [55], otherwise our proposed estimator (33) has infinite variance that renders the CRLB useless. Before we analyze this constraint, we firstly compute the following key identity

$$\begin{aligned} \mathbf{G} &= \left((\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{S}}\mathbf{U}(\mathbf{b}) \right)^\dagger (\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{S}}\mathbf{U}(\mathbf{b}) \\ &= \text{diag}(\mathbf{b}) \neq \mathbf{I}_{N_r N_t}. \end{aligned} \quad (55)$$

Then, the constraint is given [55] by

$$\mathbf{G} = \mathbf{G}I(\mathbf{h}_v)I(\mathbf{h}_v)^\dagger. \quad (56)$$

Plugging the (54) and (55) into (56), we obtain $\mathbf{G} = \mathbf{G}I(\mathbf{h}_v)I(\mathbf{h}_v)^\dagger = \mathbf{U}(\mathbf{b})$ that holds. This means that the variance of our proposed estimator is finite. Since the FIM $I(\mathbf{h}_v)$ in (54) is singular, the expression for the CRLB can be computed following [22], [55], which yields,

$$\begin{aligned} \text{CRLB}_{\text{LSE-SMP}} &\geq \mathbf{C}_{\text{LSE-SMP}} = \mathbf{G}I(\mathbf{h}_v)^\dagger \mathbf{G}^T \\ &= \sigma_n^2 \left((\bar{\mathbf{S}}\mathbf{U}(\mathbf{b}))^T \bar{\mathbf{S}}\mathbf{U}(\mathbf{b}) \right)^\dagger \end{aligned} \quad (57)$$

where $\mathbf{C}_{\text{LSE-SMP}}$ is the covariance matrix of $\hat{\mathbf{h}}_v^*$ for the LSE-SMP estimation [54]. From the CRLB theorem [50], we know that the unbiased estimator can attain the CRLB if and only if

$$\frac{\partial \ln f(\bar{\mathbf{y}}|\mathbf{h}_v)}{\partial \mathbf{h}_v} = I(\mathbf{h}_v) \left(\hat{\mathbf{h}}_v - \mathbf{h}_v \right) \quad (58)$$

always holds.

Plugging (42), (53) and (54) into (58), this verifies that our proposed LSE-SMP estimators is MVUE, and can achieve the CRLB. ■

Corollary 1: For the deterministic and sparse channel, the MSE of LSE estimator is the upper bound of that of the proposed LSE-SMP estimator. This also be denoted by $\text{MSE}_{\text{LSE-SMP}} \leq \text{MSE}_{\text{LSE}}$.

Proof: Firstly, we compute the MSE of the LSE estimator. It can be denoted by

$$\begin{aligned} \text{MSE}_{\text{LSE}} &= E\{\|\hat{\mathbf{h}}_v - \mathbf{h}_v\|^2\} \\ &= \text{trace}\{\mathbf{C}_{\text{LSE}}\} = \sum_{l=1}^{N_r N_t} [\mathbf{C}_{\text{LSE}}]_{l,l}, \end{aligned} \quad (59)$$

where $l \in \{1, 2, \dots, N_r N_t\}$. Similarly, we can obtain the MSE of the LSE-SMP estimator as follows

$$\begin{aligned} \text{MSE}_{\text{LSE-SMP}} &= E\{\|\hat{\mathbf{h}}_v^* - \mathbf{h}_v\|^2\} \\ &= \text{trace}\{\mathbf{C}_{\text{LSE-SMP}}\} = \sum_{l=1}^{N_r N_t} [\mathbf{C}_{\text{LSE-SMP}}]_{l,l}, \end{aligned} \quad (60)$$

Since the channel vector is L sparse, $\mathbf{C}_{\text{LSE-SMP}}$ has no more than L eigenvalues. Furthermore, we notice that $\mathbf{C}_{\text{LSE-SMP}}$ is obtained from the full rank matrix \mathbf{C}_{LSE} by replacing $N_r N_t - L$ rows and corresponding columns with all zero entries at each index l for which $b_l = 0$. Since the $\bar{\mathbf{S}}$ is a non-singular matrix, it is easy to prove that both \mathbf{C}_{LSE} and $\mathbf{C}_{\text{LSE-SMP}}$ are the symmetric positive definite matrices, thus their all eigenvalues are greater than zero. We denote $0 < \lambda_{N_r N_t} \leq \lambda_{N_r N_t - 1} \leq \dots \leq \lambda_1$ and $0 < \lambda_L^r \leq \lambda_{L-1}^r \leq \dots \leq \lambda_1^r$ as the eigenvalues of \mathbf{C}_{LSE} and $\mathbf{C}_{\text{LSE-SMP}}$ respectively. By applying the theorem 4.3.17 in [56] obtains

$$\lambda_1 \geq \lambda_1^r \geq \lambda_2 \geq \lambda_2^r \geq \dots \geq \lambda_L \geq \lambda_2^L \dots, \quad (61)$$

and therefore

$$\text{trace}\{\mathbf{C}_{\text{LSE}}\} = \sum_{l=1}^{N_r N_t} \lambda_l \geq \sum_{l=1}^L \lambda_l^r = \text{trace}\{\mathbf{C}_{\text{LSE-SMP}}\}. \quad (62)$$

From (62), and recalling (59) and (60), this shows that $\text{MSE}_{\text{LSE-SMP}} \leq \text{MSE}_{\text{LSE}}$. ■

B. Analysis of Iterative Evolution of LSE-SMP

In this section, we will analyze the iterative evolution performance of the LSE-SMP. We notice that it is also very difficult to analyze by using the density evolution algorithm [44], [57] directly, since it involves a big loop that contains the iterative LSE algorithm, solving Gaussian functions, and iterative SMP. As mentioned before, we also expect to rule out the influence of LSE algorithm and focus on the SMP only, then we assume that we have the knowledge of the channel vector \mathbf{h}_v . Therefore, we have the assumption that non-zero entries of \mathbf{h}_v follow the Gaussian distribution $\mathcal{N}(u_h, \sigma_h^2)$. We define that the designed training signal $\bar{\mathbf{S}}$ follows the Gaussian distribution $\mathcal{N}(0, \sigma_s^2)$. Then, we have the following theorem.

Theorem 2: Assuming that we have the knowledge of the channel vector \mathbf{h}_v , and its non-zero entries follows the Gaussian distribution $\mathcal{N}(u_h, \sigma_h^2)$, we can obtain a close-form update rule for variable nodes after any a few iterations. It denotes by

$$u_v(k+1) = l_0 + \frac{T-1}{\sqrt{4\pi u_v(k)}} \int_R \phi(l^v(k)) e^{-\frac{(l^v(k) - u_v(k))^2}{4u_v(k)}} dl^v(k), \quad (63)$$

with

$$\phi(l^v(k)) = \frac{1}{2} \left(\frac{-\alpha - a_1 \text{snr}^{-1}}{a_2 + a_1(\text{snr}^{-1} + 1)} + \frac{\alpha e^{2l^v(k)} + a_1 \text{snr}^{-1}}{a_2 + a_1 \text{snr}^{-1}} \right) - \log \sqrt{1 + \frac{a_1}{a_2 + a_1 \text{snr}^{-1}}}, \quad (64)$$

where k denotes the number of iterations, $\text{snr} = \frac{\sigma_s^2 \sigma_h^2}{\sigma_n^2}$, $\alpha = u_h^2 \sigma_h^{-2}$, $a_1 = (1 + e^{l^v(k)})^2$, and $a_2 = (N_t - 1)e^{l^v(k)}(1 + e^{l^v(k)} + \alpha)$.

Proof: Proof of the Theorem 2: see the Appendix. ■

Although it is difficult to obtain the exact solution of the mean u_v and variance σ_v^2 of $l^v(k)$, we can study and analyze the performance of SMP algorithm by simulating its Extrinsic Information Transfer (EXIT) chart [48], [58] according to the close-form expression. This allows us to take full advantage of properties of the EXIT chart to predict the performance of our proposed algorithm. It is noted that the EXIT chart in this paper is a little difference with the EXIT chart in the paper [48] and [58]. The main difference is that we take the variances of variable nodes as the extrinsic information, while the paper [48] uses the mutual information as the extrinsic information, and the paper [58] uses the MSE as the extrinsic information. In fact, the LLRs of variances message can be transformed into mutual information and MSE directly. Furthermore, we can obtain the Corollary 2 that provides a significant criterion to judge the performance of the estimator.

Corollary 2: When the LSE-SMP has achieved the convergence, the larger convergent variance σ_v^2 the variable nodes obtain, the better performance the LSE-SMP algorithm will yield.

Proof: We can obtain an estimate on the bit error rate (BER) of the LSE-SMP after an arbitrary number of iterations. The estimation of bit error probability can be approximated by the following [48]

$$P_b \approx \frac{1}{2} \text{erfc} \left(\frac{1}{\sqrt{2}} \frac{u_v}{\sigma_v} \right) = \frac{1}{2} \text{erfc} \left(\frac{\sigma_v}{2\sqrt{2}} \right), \quad (65)$$

where $\text{erfc}(x) = 1 - \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt$ is the complementary error function, and is a monotonically decreasing and continuous on $(-\infty, +\infty)$. This means that the BER will decrease with the increase of the convergent variance σ_v^2 . Therefore, the Corollary 2 is proved. ■

V. NUMERICAL RESULTS

In this section, we report the results of a detailed numerical study on the performance of our proposed LSE-SMP algorithm using the Monte-Carlo simulations. For all numerical study, we considered the channel estimation problem in a 32×64 mmWave MIMO system. The value and positions of non-zero elements in original channel vector \mathbf{h}_v were both generated by the random way, and the non-zero entries follow a Gaussian distribution $\mathcal{N}(u_h, \sigma_h^2)$. Throughout, we considered $\text{SNR} \triangleq E\{\|\bar{\mathbf{S}}\|_F^2 / \|\bar{\mathbf{n}}\|_F^2\}$ in the interval $[-10, 40\text{dB}]$; defined the coefficients of variation of \mathbf{h}_v as $\alpha = u_h^2 / \sigma_h^2$ with the $\sigma_h^2 = 10$; sparsity ratio was calculated by $\eta = \frac{L}{N_r \times N_t} 100\%$; and the performance metric was the Normalized Mean Square Error (NMSE), given by $E\{\frac{\|\hat{\mathbf{h}}_v^* - \mathbf{h}_v\|_F^2}{\|\mathbf{h}_v\|_F^2}\}$. There are 500 different channel realizations and the average results are reported.

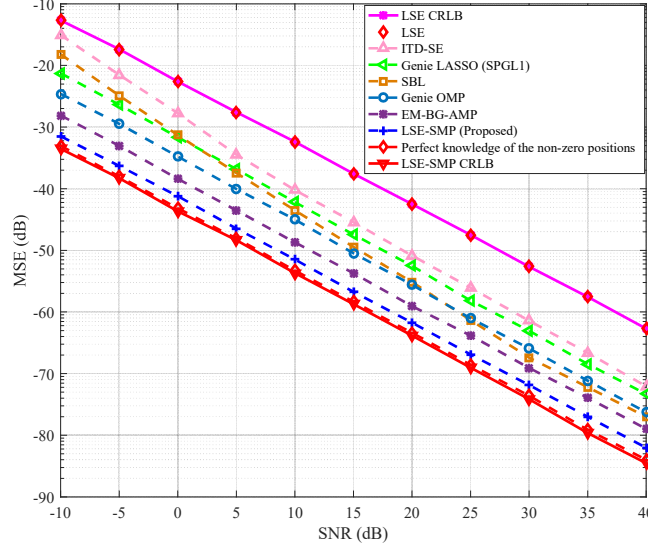


Fig. 5. This figure shows the SNR versus the average NMSE performance of various estimate algorithms. In the figure, $N_t = 32$, $N_r = 64$, $\eta = 0.7\%$, $\alpha = 10$ with the $\sigma_h^2 = 10$, and the training sequence length $T = 64$. All algorithms were run under the suggested defaults to obtain their best performance by the varied maximum number of iterations. It also shows the CRLB of LSE and the proposed LSE-SMP algorithm. The result shows the proposed LSE-SMP estimator exhibits the best NMSE performance among the tested algorithms.

A. Performance Comparison

Fig. 5 shows the average channel-estimation NMSE performance of the proposed LSE-SMP algorithm, LSE, ITD-SE [22], genie-tuned LASSO (via SPGL1 [60]), SBL [25], genie-tuned OMP [24], and EM-BG-AMP [27] (in sparse mode). All algorithms were run under the suggested defaults to obtain their best performance by the varied maximum number of iterations, with the sparse ratio at 0.7%. Additionally, we also compute the CRLB for the classical LSE and the proposed LSE-SMP estimator. The result shows that our proposed LSE-SMP estimator exhibits the best NMSE performance among the tested algorithms, and reduces the NMSE by $3dB$ relative to EM-BG-AMP that is the best of the other algorithms. As expected, the CRLB for the proposed LSE-SMP is the lowest, and this result is consistent with that of classical LSE estimator with the perfect knowledge of the non-zero positions. It is also seen from the Fig. 5 that the gap between LSE-SMP and LSE-SMP CRLB is much smaller, about $1.8dB$. This gap is partly due to the errors in the detection of the non-zero positions in sparse message passing phase and partly to the fact that all our detection strategies rely on a coarse initial estimate of the channel.

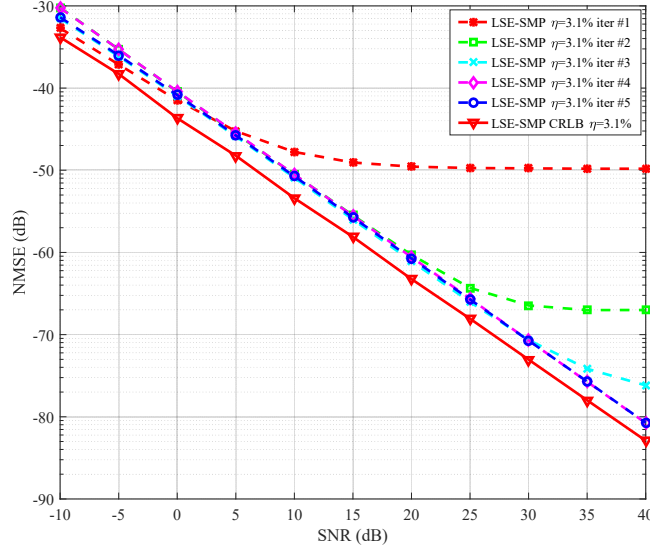


Fig. 6. This figure shows the average NMSE performance of the LSE-SMP estimator and its CRLB versus SNR under different turbo iterations. In the figure, $N_t = 32$, $N_r = 64$, $\eta = 3.1\%$, $\alpha = 10$ with the $\sigma_h^2 = 10$, and the training sequence length $T = 64$. This result shows that the LSE-SMP algorithm reaches the convergence just need 4 iterations.

B. Effect of Iterations

Fig. 6 shows the average channel-estimation NMSE performance for the LSE-SMP algorithm under several turbo iterations with sparsity ratio $\eta = 3.1\%$. The result shows that after the second turbo iteration, the NMSE performance of the LSE-SMP algorithm performs a significant improvement. Additionally, we also find that the gap of the NMSE performance between the adjacent iterations for the LSE-SMP algorithm will be smaller with the increasing of iterations. After the fourth turbo iteration, the NMSE performance have no significant improvement and it is very close to our analyzed LSE-SMP CRLB. This demonstrates that the convergence speed of the LSE-SMP algorithm is fast.

C. Effect of Sparsity Ratios

For further investigating the effect of sparsity ratio to our proposed algorithm, we fixed that the number of iterations was 5, and changed the sparsity ratio η from 0.7% to 80%. Note that different sparsity ratios η will lead to different LSE-SMP CRLBs. Simulations show that the NMSE performance of the LSE-SMP is very close to its CRLB for any of sparsity ratios. Therefore, in the Fig. 7, we just give the CRLB of the proposed LSE-SMP at the sparse ratio $\eta = 0.7\%$. We can see that the NMSE performance of the LSE is consistent with the CRLB, and it is also the upper bound of the LSE-SMP. These verify the analysis of the proposed LSE-SMP in the section IV. We also find that the NMSE performance of the LSE-SMP will be better with

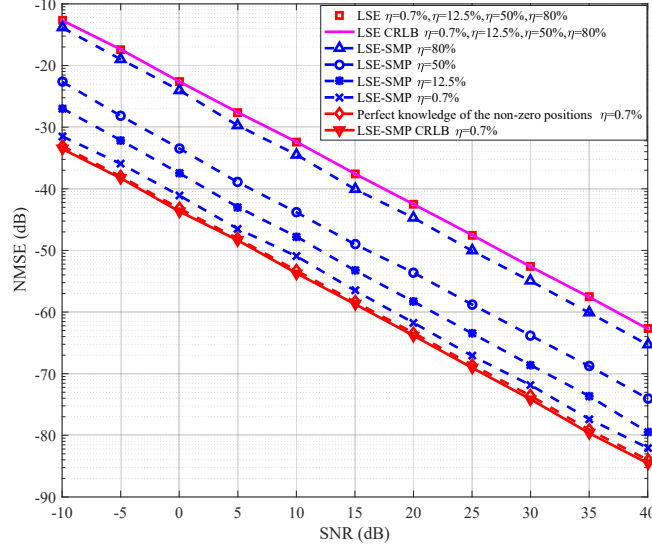


Fig. 7. This figure shows SNR versus the average NMSE performance comparison of LSE-SMP, LSE channel estimates and their CRLBs for different sparsity ratios. In the figure, $N_t = 32$, $N_r = 64$, $\eta \in \{0.7\%, 12.5\%, 50\%, 80\%\}$, $T = 64$, $\alpha = 10$ with the $\sigma_h^2 = 10$, and the iterations = 5. It can be seen that the LSE-SMP CRLB is the lower bound of the proposed LSE-SMP and the LSE CRLB is the upper bound.

the decreasing of sparse ratios as shown in the Fig. 7, mainly because the LSE-SMP is able to exploit the sparsity of the channel. To be specific, The NMSE performance of the LSE remains unchanged under different sparsity ratios. On the other hand, the NMSE performance of the LSE-SMP will decrease with the decreasing of sparsity ratios. This means that the LSE-SMP scheme will perform better especially when channel is very sparse.

D. Effect of Training Sequences Length

In Fig. 8(a), we investigate the effect of different training sequence lengths by tracking the input and output LLRs of variance messages (σ_{in}^2 and σ_{out}^2) of the variable nodes of the LSE-SMP algorithm. In this simulation, we consider the case that $N_t = 32$, $N_r = 64$, $SNR = 10dB$, $\alpha = 10$ with $\sigma_h^2 = 10$, $\eta = 12.5\%$, and $T \in \{16, 32, 64, 128, 256\}$. From the analysis of the proposed LSE-SMP in the section IV, we know that the performance of the LSE-SMP algorithm can be measured by the convergent variances of the variable nodes. Therefore, we can leverage the EXIT chart to analyse the system performance. This figure shows that the variance messages always converges to only one fixed point, and this fixed point will be higher when the training sequence length is longer. This means that the system will have lower BER with the increasing of the training sequence length. In addition, we also find that the shorter training sequence will result in the narrower space between the input and output variance traces. For example, two

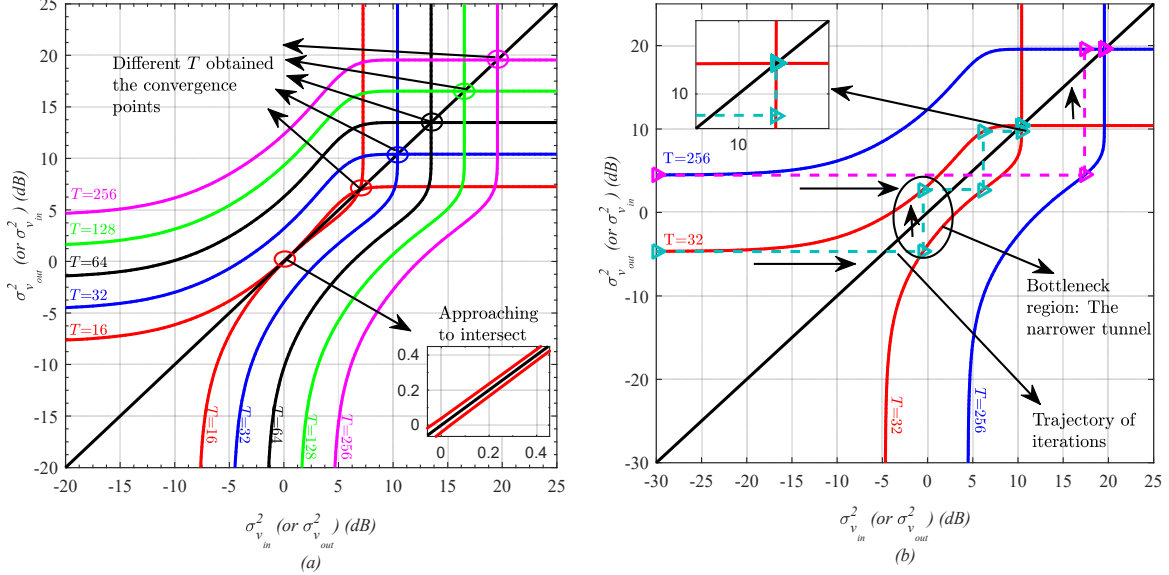


Fig. 8. This figure shows the EXIT chart analysis of the LSE-SMP algorithm, considering the case that $N_t = 32, N_t = 64, SNR = 10dB, \alpha = 10$ with $\sigma_h^2 = 10$, and $\eta = 12.5\%$. σ_{in}^2 and σ_{out}^2 are the input and output variance messages of the variable nodes in the LLRs form. (a), $T \in \{16, 32, 64, 128, 256\}$ (b), $T \in \{32, 256\}$. This figure shows that the longer training sequence not only helps the algorithm achieve a better performance, but also results in a faster convergence.

traces of σ_{in}^2 and σ_{out}^2 will approach to intersect when $T = 16$. If two traces have more one intersections, the variance message will converge the first lower intersection, which means that it will have a lower performance.

Fig. 8(b) shows the iterative trajectories of the LSE-SMP algorithm at $T = 32$ and $T = 256$ (Other simulation parameters are the same with of Fig. 8(a)). Each step of the trajectory denotes one turbo iteration in the proposed LSE-SMP algorithm. Then, It needs 6 turbo iterations to reach the convergence at $T = 32$, while it just needs 3 turbo iterations at $T = 256$. This means that the longer training sequence not only helps the algorithm achieve a better performance, but also results in a faster convergence.

E. Effect of the Coefficient of Variation of the Channel

Fig. 9 presents the effect of the coefficient of variation α of the channel vector h_v for the proposed LSE-SMP algorithm. In this simulation, we used the $N_t = 32, N_t = 64, SNR = 10dB, \sigma_h^2 = 10, \eta = 12.5\%$, and $T = 64$. Then, we plotted the EXIT chart for the coefficient of variation $\alpha \in \{0.8, 3.2, 12.8, 51.2\}$. We can see that there is only one convergent variance point for the different coefficients of variation, and the convergent variance point will be higher with the increasing of the coefficient of variation, while the *bottleneck region* that denotes the narrower region between the σ_{in}^2 and σ_{out}^2 traces keeps unchanged. These means that the more

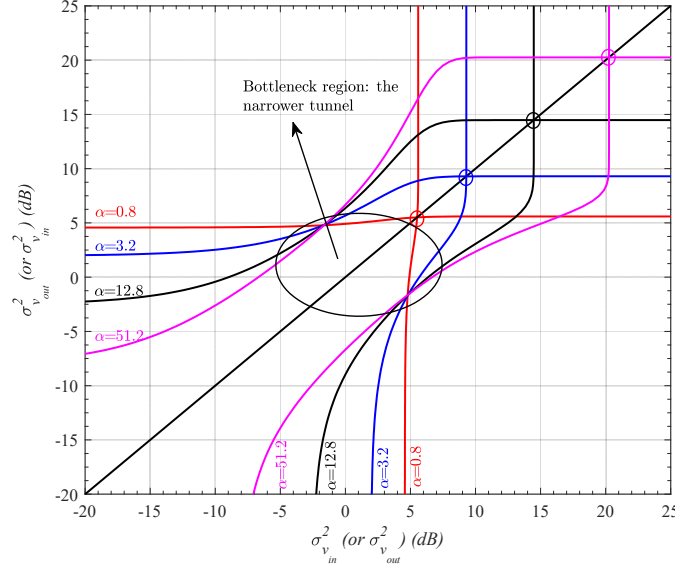


Fig. 9. This figure shows the EXIT chart analysis of the LSE-SMP algorithm with $T = 64$, $N_t = 32$, $N_t = 64$ and $SNR = 10dB$, where we set the coefficients of variation $\alpha \in \{0.8, 3.2, 12.8, 51.2\}$ under the $\sigma_h^2 = 10$. The result shows that the more dispersion channel condition will lead to the lower system performance, while speed the less time to achieve the convergence.

dispersion channel condition will lead to the lower system performance, while maybe speeds the less time to achieve the convergence.

F. Complexity of the Algorithms

Fig. 10 shows the NMSE versus the runtime of various algorithms. We evaluate the runtime of each algorithm on a typical personal computer, where we set $N_t = 32$, $N_t = 64$, $SNR = 40dB$, $\alpha = 10$ with $\sigma_h^2 = 10$, $\eta = 0.7\%$, and $T = 64$. Then, by varying the maximum number of iterations in ITD-SE, SBL, OMP, EM-BG-AMP, LSE-SMP algorithms, we obtained their NMSE-runtime frontier. The other two algorithms are represented by two points. We notice that the ITD-SE obtains the best performance-complexity trade-off when the NMSE is larger than $-66dB$, LASSO gives the best trade-off when the NMSE is between $-66dB$ and $-73dB$, EM-BG-AMP gives the best trade-off when the NMSE is between $-73dB$ and $-77dB$, and LSE-SMP is the best when the NMSE is less than $-77dB$. In other words, the proposed LSE-SMP algorithm obtains the best performance than other tested algorithms, although it spends more time.

VI. CONCLUSION

In this paper, a novel channel estimation algorithm for mmWave MIMO systems was proposed, which leverages both virtues of the SMP and LSE algorithms. We analyzed the CRLB of the proposed LSE-SMP algorithm, and showed that the algorithm was MVUE. Next, we also presented a

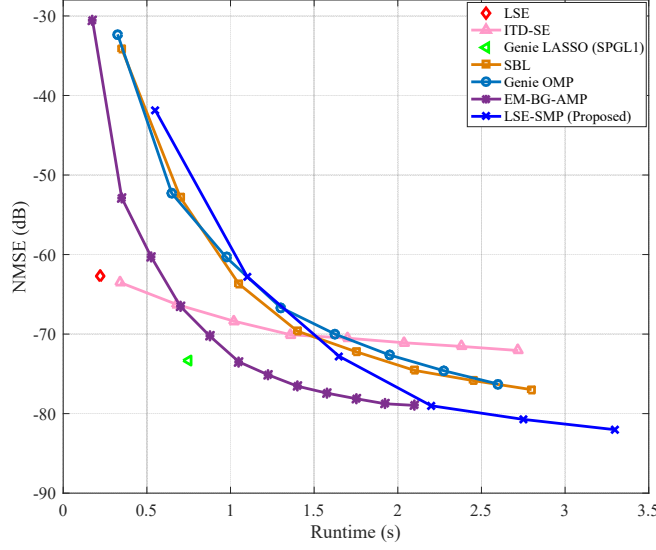


Fig. 10. This figure shows the NMSE versus the runtime of various algorithms. In the figure, $N_t = 32, N_r = 64, SNR = 40dB, \alpha = 10$ with $\sigma_h^2 = 10, \eta = 0.7\%$, and $T = 64$. This result indicates that the proposed LSE-SMP obtains the best performance-complexity trade-off than the tested algorithms when the NMSE is less than $-77dB$.

powerful analysis technique, which can provide significant insights on the convergence behavior on the iteration evolution of the proposed algorithm. Simulation experiments verified that the proposed algorithm reduced the NMSE by $3dB$ relative to the best of existing algorithms. In addition, it was also shown that the proposed algorithm typically needed only 5 turbo iterations to achieve CRLB. However, one limitation in our paper is the assumption that the virtual channel vector \mathbf{h}_v is L -sparse. In the future work, we will relax the assumption and investigate the behaviour of the proposed estimator if the L is overestimated or underestimated case.

APPENDIX

Since we obtained the LLRs form of the LSE-SMP iteration process, many previous research results show that (40) can be well approximated by Gaussian densities [44]. we denote the mean and variance of $l_{ij \rightarrow i\tau}^v$ by u_v and σ_v^2 . There is an important condition, called the symmetry condition. For a Gaussian signal with the mean u_v and variance σ_v^2 , this condition reduces to $\sigma_v^2 = 2u_v$ [40], [44], [59], which means that we only need to keep the mean. Then, (40) simply becomes

$$\begin{aligned}
 E\{l_{ij \rightarrow i\tau}^v(k+1)\} &= u_v(k+1) = E\left\{l_0 + \sum_{t \neq \tau}^T l_{it \rightarrow ij}^s(k)\right\} \\
 &= E\left\{l_0 + \sum_{t \neq \tau}^T \log \frac{f(y_{it}|e_{it \rightarrow ij}^s(k) + s_{tj}\hat{h}_{ij}(k), v_{it \rightarrow ij}^s(k) + s_{tj}^2 v_{h_{ij}}(k))}{f(y_{it}|e_{it \rightarrow ij}^s(k), v^s(k)_{it \rightarrow ij}(k))}\right\}. \tag{66}
 \end{aligned}$$

Plugging (24) into (66), and we obtain the update as following,

$$u_v(k+1) = l_0 + E \left\{ \sum_{t \neq \tau}^T \frac{(y_{i\tau} - e_{i\tau \rightarrow ij}^s(k))^2}{2(v_{i\tau \rightarrow ij}^s(k))} - \frac{(y_{i\tau} - e_{i\tau \rightarrow ij}^s(k) - s_{\tau j} \hat{h}_{ij}(k))^2}{2(v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k))} - \log \sqrt{\frac{v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k)}{v_{i\tau \rightarrow ij}^s(k)}} \right\} \quad (67)$$

In order to be convenient for analyse, we define

$$\phi(l_{ij \rightarrow i\tau}^v(k)) = \frac{(y_{i\tau} - e_{i\tau \rightarrow ij}^s(k))^2}{2(v_{i\tau \rightarrow ij}^s(k))} - \frac{(y_{i\tau} - e_{i\tau \rightarrow ij}^s(k) - s_{\tau j} \hat{h}_{ij}(k))^2}{2(v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k))} - \log \sqrt{\frac{v_{i\tau \rightarrow ij}^s(k) + s_{\tau j}^2 v_{h_{ij}}(k)}{v_{i\tau \rightarrow ij}^s(k)}}. \quad (68)$$

Due to $s_{tj} \sim \mathcal{N}(0, \sigma_s^2)$, we can obtain $E(s_{tj}^2) = Var(s_{tj}) + E(s_{tj})^2 = \sigma_s^2$. Similarly, we can get $E(n_{i\tau}^2) = Var(n_{i\tau}) + E(n_{i\tau})^2 = \sigma_n^2$. Plugging (37)-(39) into the above expression, we obtain

$$\begin{aligned} \phi(l_{ij \rightarrow i\tau}^v(k)) = & -\log \left(\sqrt{\frac{\sum_{m \neq j} \frac{s_{\tau m}^2}{1+e^{-l_{ij \rightarrow i\tau}^v(k)}} (\sigma_h^2 + \frac{u_h^2}{1+e^{l_{ij \rightarrow i\tau}^v(k)}}) + s_{\tau j}^2 \sigma_h^2 + \sigma_n^2}{\sum_{m \neq j} \frac{s_{\tau m}^2}{1+e^{-l_{ij \rightarrow i\tau}^v(k)}} (\sigma_h^2 + \frac{u_h^2}{1+e^{l_{ij \rightarrow i\tau}^v(k)}}) + \sigma_n^2}} \right) + \\ & \frac{-\left(\frac{s_{\tau j} \hat{h}_{ij}(k)}{1+e^{-l_{ij \rightarrow i\tau}^v(k)}} - s_{\tau j} \hat{h}_{ij}(k) + n_{i\tau}\right)^2}{2 \sum_{m \neq j} \frac{s_{\tau m}^2}{1+e^{-l_{ij \rightarrow i\tau}^v(k)}} (\sigma_h^2 + \frac{u_h^2}{1+e^{l_{ij \rightarrow i\tau}^v(k)}}) + s_{\tau j}^2 \sigma_h^2 + \sigma_n^2}} + \frac{\left(\frac{s_{\tau j} \hat{h}_{ij}(k)}{1+e^{-l_{ij \rightarrow i\tau}^v(k)}} + n_{i\tau}\right)^2}{2 \sum_{m \neq j} \frac{s_{\tau m}^2}{1+e^{-l_{ij \rightarrow i\tau}^v(k)}} (\sigma_h^2 + \frac{u_h^2}{1+e^{l_{ij \rightarrow i\tau}^v(k)}}) + \sigma_n^2}}. \end{aligned} \quad (69)$$

Since transmit and receive antennas are independent and symmetric, we omit the subscript of $l_{ij \rightarrow i\tau}^v(k)$, and use $l^v(k)$ to denote the $l_{ij \rightarrow i\tau}^v(k)$ in the following paper. Then, we have

$$\begin{aligned} \phi(l^v(k)) = & -\log \left(\sqrt{\frac{\frac{(N_t-1)\sigma_s^2}{1+e^{-l^v(k)}} (\sigma_h^2 + \frac{u_h^2}{1+e^{l^v(k)}}) + \sigma_s^2 \sigma_h^2 + \sigma_n^2}{\frac{(N_t-1)\sigma_s^2}{1+e^{-l^v(k)}} (\sigma_h^2 + \frac{u_h^2}{1+e^{l^v(k)}}) + \sigma_n^2}} \right) + \\ & \frac{\frac{-\sigma_s^2 u_h^2}{(1+e^{l^v(k)})^2} - \sigma_n^2}{2 \left(\frac{(N_t-1)\sigma_s^2}{1+e^{-l^v(k)}} (\sigma_h^2 + \frac{u_h^2}{1+e^{l^v(k)}}) + \sigma_s^2 \sigma_h^2 + \sigma_n^2 \right)} + \frac{\frac{\sigma_s^2 u_h^2}{(1+e^{-l^v(k)})^2} + \sigma_n^2}{2 \left(\frac{(N_t-1)\sigma_s^2}{1+e^{-l^v(k)}} (\sigma_h^2 + \frac{u_h^2}{1+e^{l^v(k)}}) + \sigma_n^2 \right)}. \end{aligned} \quad (70)$$

To simplify the expression, we yield

$$\begin{aligned} \phi(l^v(k)) = & -\log \left(\sqrt{1 + \frac{(1+e^{l^v(k)})^2}{(N_t-1)e^{l^v(k)}(1+e^{l^v(k)}+u_h^2\sigma_h^{-2})+(1+e^{l^v(k)})^2snr-1}} \right) + \frac{1}{2} \\ & \left(\frac{-u_h^2\sigma_h^{-2}-(1+e^{l^v(k)})^2snr-1}{(N_t-1)e^{l^v(k)}(1+e^{l^v(k)}+u_h^2\sigma_h^{-2})+(1+e^{l^v(k)})^2(sn-1+1)} + \frac{u_h^2\sigma_h^{-2}e^{2l^v(k)}+(1+e^{l^v(k)})^2snr-1}{(N_t-1)e^{l^v(k)}(1+e^{l^v(k)}+u_h^2\sigma_h^{-2})+(1+e^{l^v(k)})^2snr-1} \right), \end{aligned} \quad (71)$$

where $snr = \frac{\sigma_s^2 \sigma_h^2}{\sigma_n^2}$. The above expression looks like complex, however, we find that it is symmetric and has a few common terms. Then, we can define $\alpha = u_h^2 \sigma_h^{-2}$, $a_1 = (1 + e^{l^v(k)})^2$,

and $a_2 = (N_t - 1)e^{l^v(k)}(1 + e^{l^v(k)} + \alpha)$. The $\phi(l^v(k))$ can be denoted by

$$\phi(l^v(k)) = \frac{1}{2} \left(\frac{-\alpha - a_1 \text{snr}^{-1}}{a_2 + a_1(\text{snr}^{-1} + 1)} + \frac{\alpha e^{2l^v(k)} + a_1 \text{snr}^{-1}}{a_2 + a_1 \text{snr}^{-1}} \right) - \log \sqrt{1 + \frac{a_1}{a_2 + a_1 \text{snr}^{-1}}}, \quad (72)$$

Since l^v is Gaussian with the mean u_v and variance $2u_v$, we obtain the following update expression by the definition of the expectation

$$u_v(k+1) = l_0 + \frac{T-1}{\sqrt{4\pi u_v(k)}} \int_R \phi(l^v(k)) e^{-\frac{(l^v(k)-u_v(k))^2}{4u_v(k)}} dl^v(k). \quad (73)$$

Therefore, we have the Theorem 2.

REFERENCES

- [1] C. W. Huang, L. Liu, C. Yuen, S. M. Sun, "A LSE and Sparse Message Passing-Based Channel Estimation for mmWave MIMO Systems," will be appeared in *Proc. of the 2016 IEEE Global Communications (GlobeCom) Conference*, Washington, DC USA, Dec. 2016.
- [2] T. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. Wong, J. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5G cellular: It will work," *IEEE Access*, vol. 1, pp. 335C349, 2013.
- [3] M. Akdeniz, Y. Liu, M. Samimi, S. Sun, S. Rangan, T. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1164C1179, June 2014.
- [4] T. S. Rappaport, R. W. Heath Jr., R. C. Daniels, and J. Murdock, "Millimeter Wave Wireless Communications". *PrenticeHall*, Sep. 2014.
- [5] X. Gao, L. Dai, A. Sayeed, "Low RF-Complexity Technologies for 5G Millimeter-Wave MIMO Systems with Large Antenna Arrays", arXiv preprint arXiv:1607.04559, 2016
- [6] R. W. Heath Jr., N. G. Prelcic, S. Rangan, W. Roh, and A. Sayeed, "An Overview of Signal Processing Techniques for Millimeter Wave MIMO Systems," to appear in *IEEE Journal of Selected Topics in Signal Processing*, Apr. 2016.
- [7] T. S. Rappaport, G. R. MacCartney, Jr., M. K. Samimi, and S. Sun, "Wideband millimeter-wave propagation measurements and channel models for future wireless communication system design" *IEEE Trans. on Commun.*, vol. 63, no. 9, pp. 3029-3056, Sept. 2015.
- [8] S. Sun, G. R. MacCartney, Jr., and T. S. Rappaport, "Millimeter-Wave Distance-Dependent Large-Scale Propagation Measurements and Path Loss Models for Outdoor and Indoor 5G Systems," in *the 10th European Conference on Antennas and Propagation (EuCAP 2016)*, Apr. 2016.
- [9] Q. Xue, X. Fang, M. Xiao and L. Yan, "Multi-user Millimeter Wave Communications with Nonorthogonal Beams," in *IEEE Trans. on Veh. Technol.*, vol. PP, no.99, pp.1-1 doi: 10.1109/TVT.2016.2617083.
- [10] O. El Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. W. Heath Jr., "Spatially sparse precoding in millimeter wave MIMO systems," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 3, pp. 1499C1513, Mar. 2014.
- [11] A. M. Sayeed, "Deconstructing multi-antenna fading channels," *IEEE Trans. on Signal Processing*, pp. 2563C2579, Oct. 2002.
- [12] P. Schniter and A. M. Sayeed, "A Sparseness-Preserving Virtual MIMO Channel Model," in *Proc. Conf. on Information Sciences and Systems*, (Princeton, NJ), pp. 36-41, Mar. 2004.
- [13] P. Schniter and A. Sayeed, "Channel Estimation and Precoder Design for Millimeter-Wave Communications: The Sparse Way," in *Proc. Asilomar Conf. on Signals, Systems, and Computers*, Nov. 2014

- [14] A. Ghosh, T. Thomas, M. Cudak, R. Ratasuk, P. Moorut, F. Vook, T. Rappaport, G. Maccartney, S. Sun, and S. Nie, "Millimeter-wave enhanced local area systems: A high-data-rate approach for future wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp.1152C1163, June 2014
- [15] H. Niu, M. Shen, J. A. Ritcey and H. Liu, "A factor graph approach to iterative channel estimation and LDPC decoding over fading channels," in *IEEE Trans. on Wireless Commun.*, vol. 4, no. 4, pp. 1345-1350, July 2005.
- [16] M. F. Flanagan and A. D. Fagan, "Iterative Channel Estimation, Equalization, and Decoding for Pilot-Symbol Assisted Modulation Over Frequency Selective Fast Fading Channels," in *IEEE Trans. on Veh. Technol.*, vol. 56, no. 4, pp. 1661-1670, July 2007.
- [17] Q. H. Guo, L. Ping and D. Huang, "A low-complexity iterative channel estimation and detection technique for doubly selective channels" in *IEEE Trans. on Wireless Commun.*, vol. 8, no. 8, pp. 4340-4349, August 2009.
- [18] Y. Zhu, D. Guo and M. L. Honig, "A message-passing approach for joint channel estimation, interference mitigation, and decoding," *IEEE Trans. on Wireless Commun.*, vol. 8, no. 12, pp. 6008-6018, Dec. 2009.
- [19] S. Park, B. Shim and J. W. Choi, "Iterative Channel Estimation Using Virtual Pilot Signals for MIMO-OFDM Systems," in *IEEE Trans. on Signal Process.*, vol. 63, no. 12, pp. 3032-3045, June15, 2015.
- [20] S. Wu, L. L. Kuang, Z. Y. Ni, D. Huang, Q. H. Guo and J. H. Lu, "Message-Passing Receiver for Joint Channel Estimation and Decoding in 3D Massive MIMO-OFDM Systems," submitted to *IEEE Trans. on Wireless Commun.*, Jan. 2016.
- [21] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing: I. Motivation and construction" in *Proc. of Inform. Theory Workshop*, (Cairo, Egypt), Jan. 2010.
- [22] C. Carbonelli, S. Vedantam and U. Mitra, "Sparse Channel Estimation with Zero Tap Detection," *IEEE Trans. on Wireless Commun.*, vol. 6, no. 5, pp. 1743-1763, May 2007.
- [23] A. Alkhateeb, O. El Ayach, G. Leus and R. W. Heath, "Channel Estimation and Hybrid Precoding for Millimeter Wave Cellular Systems," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 831-846, Oct. 2014.
- [24] J. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Info. Theory*, vol. 53, pp. 4655-4666, Dec. 2007.
- [25] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, pp. 2153-2164, Aug. 2004.
- [26] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc. B*, vol. 58, no. 1, pp. 267-288, 1996.
- [27] J. P. Vila and P. Schniter, "Expectation-Maximization Gaussian-Mixture Approximate Message Passing," in *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4658-4672, Oct.1, 2013.
- [28] J. Mo, P. Schniter, N. Gonzalez-Prelcic, and R. W. Heath, Jr., "Channel estimation in millimeter wave MIMO systems with one-bit quantization," in *Proc. Asilomar Conf. on Signals, Systems and Computers*, Nov. 2014.
- [29] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. New York: Cambridge University Press, 2005.
- [30] V. Raghavan and A. Sayeed, "Sublinear capacity scaling laws for sparse MIMO channels," *IEEE Trans. Inf. Theory*, vol. 57, no. 1, pp.345C364, Jan 2011.
- [31] A. Alkhateeb and R. W. Heath, "Frequency Selective Hybrid Precoding for Limited Feedback Millimeter Wave Systems," in *IEEE Trans. on Com.*, vol. 64, no. 5, pp. 1801-1818, May 2016.
- [32] C. Gustafson, K. Haneda, S. Wyne, F. Tufvesson, "On mm-wave multipath clustering and channel modeling", *IEEE Trans. Antennas Propag.*, vol. 62, no. 3, pp. 1445-1455, Mar. 2014.
- [33] L. Dai, X. Gao, X. Su, S. Han, C.-L. I, and Z. Wang, "Low-complexity soft-output signal detection based on Gauss-Seidel method for uplink multi-user large-scale MIMO systems," *IEEE Trans. Veh. Technol.*, vol. PP, 2015.
- [34] J. Brady, N. Behdad, and A. M. Sayeed, "Beamspace MIMO for millimeter-wave communications: System architecture, modeling, analysis, and measurements," *IEEE Trans. on Antennas and Propagation*, vol. 61, pp. 3814-3827, July 2013.

- [35] A. Sayeed and J. Brady, "High Frequency Differential MIMO: Basic Theory and Transceiver Architectures", in *Proc. of IEEE ICC*, London, June 2015.
- [36] X. Gao, L. Dai, A. Sayeed, "Beamspace Channel Estimation for Millimeter-Wave Massive MIMO Systems with Lens Antenna Array", arXiv preprint arXiv:1607.05130, 2016.
- [37] H. A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping and F. R. Kschischang, "The Factor Graph Approach to Model-Based Signal Processing" in *Proceedings of the IEEE*, vol. 95, no. 6, pp. 1295-1322, June 2007.
- [38] M. Nassar, P. Schniter, and B. Evans, "A Factor-Graph Approach to Joint OFDM Channel Estimation and Decoding in Impulsive Noise Environments," *IEEE Trans. on Signal Processing*, vol. 62, no. 6, pp. 1576-1589, Mar. 2014.
- [39] L. Liu, C. Yuen, Y. L. Guan, Y. Li and Y. Su, "Convergence Analysis and Assurance for Gaussian Message Passing Iterative Detector in Massive MU-MIMO Systems," in *IEEE Trans. on Wireless Commun.*, vol. 15, no. 9, pp. 6487-6501, Sept. 2016.
- [40] T. J. Richardson and R. Urbanke, "The capacity of low-density parity check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, pp. 599-618, Feb. 2001.
- [41] Q. Guo and L. Ping, "LMMSE turbo equalization based on factor graphs" *IEEE J. Sel. Areas Commun.*, vol. 26, no. 2, pp. 311-319, 2008.
- [42] T. L. Narasimhan and A. Chockalingam, "Channel Hardening-Exploiting Message Passing (CHEMP) Receiver in Large-Scale MIMO Systems," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 847-860, Oct. 2014.
- [43] L. Liu, C. Yuen, Y. L. Guan, Y. Li and Y. P. Su, "A Low-Complexity Gaussian Message Passing Iterative Detection for Massive MU-MIMO Systems," in *Proc IEEE International Conference on Information, Communications and Signal Processing (ICICS)*, Singapore, Dec. 2015.
- [44] S.-Y. Chung, T. J. Richardson and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," in *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 657-670, Feb 2001.
- [45] Q. Su and Y. C. Wu, "Convergence analysis of the variance in Gaussian belief propagation" *IEEE Trans. Signal Process.*, vol. 62, no. 19, pp. 5119-5131, Oct. 2014.
- [46] L. Ping, L. Liu, K. Wu, and W. K. Leung, "Interleave Division Multiple-Access," *IEEE Trans. Wireless Commun.*, vol. 5, no. 4, pp. 938-947, Apr. 2006.
- [47] L. Liu, Y. Li, Y. Su, and Y. Sun, "Quantize-and-Forward Strategy for Interleave-Division Multiple-Access Relay Channel," *IEEE Trans. on Veh. Technol.* 65(3): 1808-1814, 2016.
- [48] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727-1737, Oct. 2001.
- [49] A. Van den Bos, *Parameter estimation for scientists and engineers*. John Wiley and Sons, 2007.
- [50] Steven M. Kay., *Fundamentals of statistical signal processing: Estimation Theory*. 1st edition. Upper Saddle River, NJ: Prentice Hall PTR, 1993.
- [51] U. Niesen, D. Shah and G. W. Wornell, "Adaptive Alternating Minimization Algorithms," in *IEEE Trans. Inf. Theory*, vol. 55, no. 3, pp. 1423-1429, March 2009.
- [52] R. Niazadeh, M. Babaie-Zadeh, and C. Jutten, "An alternating minimization method for sparse channel estimation," in *Proc. 9th Int. Conf. Latent Variable Anal. Signal Separation*, 2010, pp. 319C327.
- [53] X. Lu, T. Gong, P. Yan, Y. Yuan and X. Li, "Robust Alternative Minimization for Matrix Completion," in *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 42, no. 3, pp. 939-949, Jun. 2012.
- [54] Couillet, R., F. Pascal, and J. W. Silverstein, "Robust Estimates of Covariance Matrices in the Large Dimensional Regim", *IEEE Trans. Inf. Theory*, vol. 60, issue 11, pp. 7269-7278, 2014.

- [55] P. Stoica and T. L. Marzetta, "Parameter estimation problems with singular information matrices," *IEEE Trans. Signal Processing*, vol. 49, no. 1, pp. 87-89, Jan. 2001.
- [56] R. A. Horn, C. R. Johnson, *Matrix Analysis*, 2nd ed, 2013, Cambridge Univ. Press.
- [57] A. Maleki, "Coherence analysis of iterative thresholding algorithms," , *47th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, 2009, pp. 236-243.
- [58] K. Bhattad and K. R. Narayanan, "An MSE-based transfer chart for analyzing iterative decoding schemes using a Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 53, no. 1, pp. 22-38, Jan. 2007.
- [59] T. J. Richardson, M. A. Shokrollahi and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," in *IEEE Transactions on Inf. Theory*, vol. 47, no. 2, pp. 619-637, Feb 2001.
- [60] E. van den Berg and M. P. Friedlander, "Probing the Pareto frontier for basis pursuit solutions," *SIAM J. Scientif. Comput.*, vol. 31, no. 2, pp. 890-912, 2008.